

A Retrieval and Display System for the Holy Quran

by

Jamil Izzat J. Ahmad

A Thesis Presented to the

FACULTY OF THE COLLEGE OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

COMPUTER SCIENCE

June, 1988

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA
313/761-4700 800/521-0600

A RETRIEVAL AND DISPLAY SYSTEM FOR THE HOLY
QURAN

BY
JAMIL IZZAT J. AHMAD

A Thesis Presented to the
FACULTY OF THE COLLEGE OF GRADUATE STUDIES
KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN, SAUDI ARABIA

LIBRARY
KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
Dhahran - 31261, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE
In
COMPUTER SCIENCE

JUNE 1988

UMI Number: 1381100

UMI Microform 1381100
Copyright 1996, by UMI Company. All rights reserved.

**This microform edition is protected against unauthorized
copying under Title 17, United States Code.**

UMI
300 North Zeeb Road
Ann Arbor, MI 48103

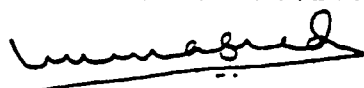
**KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS
DHAHRAN 31261, SAUDI ARABIA**

COLLEGE OF GRADUATE STUDIES

This thesis, written by JAMIL IZZAT JAMIL AHMAD under the direction of his Thesis Advisor and approved by his Thesis Committee, has been presented to and accepted by the Dean of the College of Graduate studies, in partial fulfillment of the requirements for the degree of MASTER OF SCIENCE in COMPUTER SCIENCE.

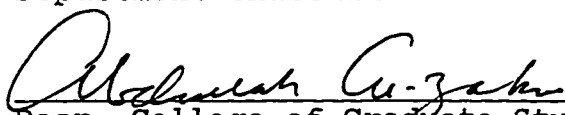
THESIS COMMITTEE

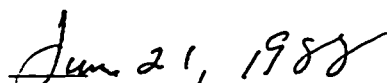

Thesis Advisor

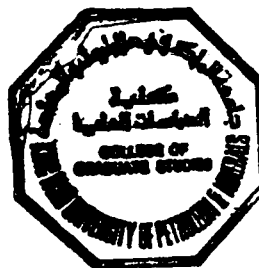

Member

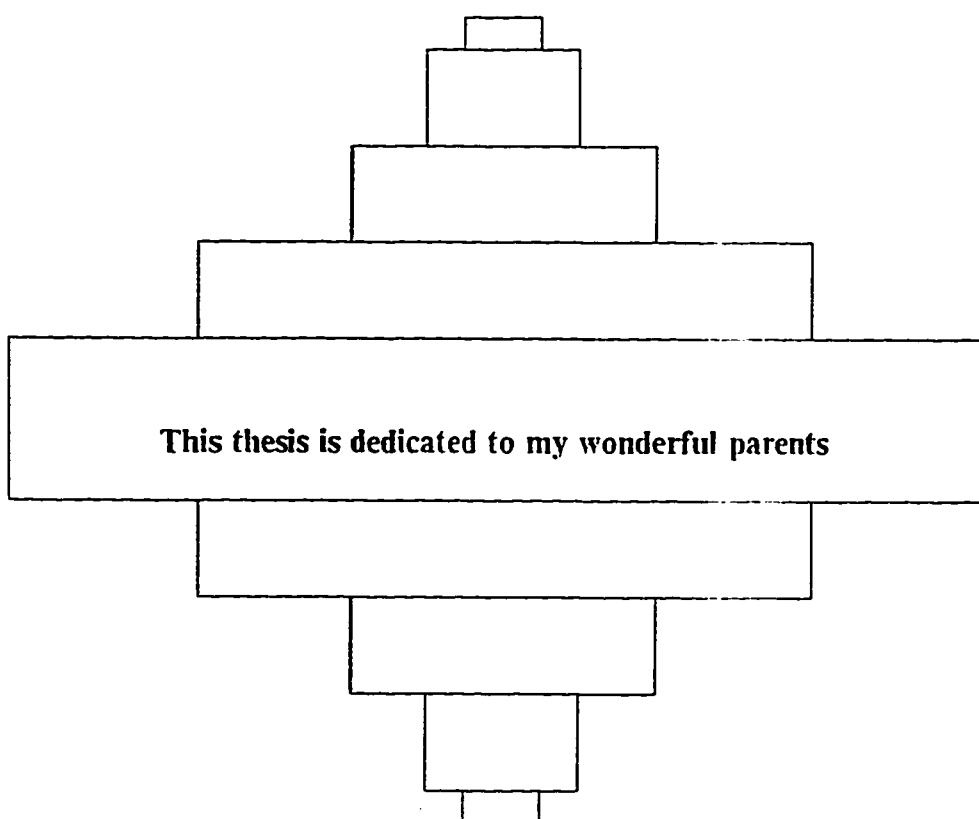

Member


Department Chairman


Dean, College of Graduate Studies


Date





This thesis is dedicated to my wonderful parents

ACKNOWLEDGMENT

Acknowledgment is due to the King Fahd University of Petroleum and Minerals for providing the opportunity to carry out this research work.

I gratefully acknowledge the tremendous help, guidance and encouragement by Dr. Mohammad Ghazaly Khayyat who served as my committee chairman. I express my thanks and appreciation to the other members of my thesis Committee, Dr. Manzer Masud and Dr. Abdullah Al-Mashooqi.

I gratefully thank my father and sister for their help in the data entry of the Holy Quran. I wish to express special thanks and appreciation to Mr. Anan Yaagoub who helped me in writing the thesis. I especially thank Mr. Husni Al-Muhtasib for his valuable help in ideas, encouragement, and time. Finally, I thank everybody who supported me by any means.

TABLE OF CONTENTS

	Page
LIST OF TABLES.....	ix
LIST OF FIGURES.....	x
ABSTRACT (English).....	xi
ABSTRACT (Arabic).....	xii
1. INTRODUCTION.....	1
2. EXISTING SYSTEMS.....	3
2.1 Salsabeel.....	3
2.2 Al-Raed.....	4
2.3 Al-Alamiah.....	5
3. DESIGN AND IMPLEMENTATION.....	8
3.1 User Interface Module.....	12
3.2 Graphics And Contextual Analysis Module...	13
3.2.1 Graphics Module.....	13
3.2.1.1 Shape Generation.....	14
3.2.1.1.1 New Shape Generation.....	14
3.2.1.1.2 Old Shape Modification.....	15
3.2.1.2 Shape Drawing.....	16
3.2.2 Contextual Analysis Module.....	18
3.2.2.1 Contextual Analysis For Data Reading.....	19
3.2.2.1.1 Class Determination.....	20

3.2.2.1.2 Shape Size Determination...	20
3.2.2.1.3 Shape Erase.....	20
3.2.2.2 Data Drawing Contextual	
Analysis.....	24
3.2.3 Other Related Routines.....	29
3.2.3.1 Number Drawing.....	30
3.2.3.2 Number Reading.....	30
3.2.3.3 Phrase Display.....	32
3.2.3.4 Menu Border Drawing.....	32
3.2.3.5 Appendix Display.....	33
3.2.3.6 Text Printing.....	33
3.3 Storage Module.....	34
3.3.1 Random Access Files.....	35
3.3.2 Text Files.....	35
3.3.3 Comparison Of The Choices.....	36
3.3.3.1 Search For A Specific Sora.....	36
3.3.3.2 Search For A Specific Aiah In A	
Sora.....	37
3.3.3.3 Search For A Juz.....	37
3.3.3.4 Search For A Hezb.....	37
3.3.3.5 Search For A Word Or Phrase.....	37
3.3.3.6 Display Speed.....	38
3.3.3.7 Space Consideration.....	38
3.3.3.8 Conclusion.....	39

3.3.4 Advantages Of The Design.....	39
3.4 Data Entry Module.....	40
3.4.1 The Main Procedure.....	40
3.4.2 File Determination.....	42
3.4.3 Data Addition (Storage).....	42
3.4.4 Deletion Of A Juz.....	45
3.4.5 Modification Of A Juz.....	46
3.4.5.1 Read Text.....	46
3.5 Search Module.....	50
3.5.1 The Main Routine.....	52
3.5.2 Read Text.....	52
3.5.3 Initialize Tables.....	53
3.5.4 Sora And Aiah Search.....	53
3.5.5 Juz Search.....	58
3.5.6 Hezb Search.....	61
3.5.7 Word And Phrase Search.....	64
3.5.7.1 The Main Procedure.....	68
3.5.7.2 Sequential Search Routine.....	69
3.5.7.3 Direct Search Routine.....	73
3.5.8 Display Text.....	77
3.5.8.1 Frame Display.....	78
3.5.8.2 Display Information Inside The Frame.....	78
3.5.8.3 Display A Line.....	81

3.5.8.4 Display Until End-Of-Juz.....	82
3.5.8.5 Display Until End-Of-Quran.....	84
3.6 Tajweed Rules Module.....	86
3.6.1 The Main Procedure.....	88
3.6.2 Display Information About Tajweed Rules.....	90
3.6.3 Aiah Analysis.....	90
3.6.3.1 The Main Procedure.....	91
3.6.3.2 Display Aiah.....	92
3.6.3.3 Get A Word.....	93
3.6.3.4 Display A Rule.....	94
3.6.3.5 Rules Analysis.....	94
4. CONCLUSION AND FURTHER WORK.....	99
REFERENCES.....	102
APPENDIX A.....	103
APPENDIX B.....	129

LIST OF TABLES

	Page
1 ASCII Table.....	17

LIST OF FIGURES

	Page
1 Block Diagram Of The System.....	10
2 Block Diagram Of The Contextual Analysis For Data Reading Routine.....	21
3 Block Diagram Of The Data Drawing Contextual Analysis Routine.....	26
4 Block Diagram Of The Data Entry Routine.....	41
5 Sora Table.....	56
6 Juz Table.....	60
7 Hezb Table.....	63
8 Word Table.....	66
9 Block Diagram Of The Word-And-Phrase-Search Routine.....	67
10 Block Diagram Of The Display Text Routine....	79
11 Block Diagram Of Tajweed Rules Module.....	89

THESIS ABSTRACT

NAME OF STUDENT: JAMIL IZZAT JAMIL AHMAD

TITLE OF STUDY : A RETRIEVAL AND DISPLAY SYSTEM
FOR THE HOLY QURAN

MAJOR FIELD : COMPUTER SCIENCE AND ENGINEERING

DATE OF DEGREE : JUNE 1988

Since the shine of Islam, Muslims have been taking care of their Holy Book, the Quran. Today, as we live in the computer and information age, a computerized system for the Holy Quran is very useful . This system can be utilized in education, research concerned with the Holy Quran, and personal use.

The main objective of this research is to develop a retrieval and display system for the Holy Quran. The system displays and prints the Holy Quran text using Al-Rasm Al-Othmani style. It Provides a multi-index search for a Sora, Aiah, Hezb, Juz, word, or phrase. In addition, it analyzes any specific Aiah for Tajweed rules. It is a modular and flexible system,i.e., efficient for additions and modifications. Furthermore, it is menu-driven and fully Arabic. The system is implemented on the IBM Personal Computers (PC) and the compatibles which are the most popular machines.

MASTER OF SCIENCE DEGREE

KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS

Dhahran, Saudi Arabia

June 1988

- xi -

خلاصة الرسالة

اسم الطالب : جميل عزت جميل أحمد
عنوان الدراسة : نظام لعرض وتخزين واسترجاع القرآن الكريم
التخصص : علوم وهندسة الحاسب الآلي
تاريخ الشهادة : يونيو ١٩٨٨ م

لقد اعتنى المسلمون بكتابهم المقدس القرآن الكريم بدرجة مميزة منذ بزوغ فجر الاسلام . وفي الوقت الحالي ومع تطور مجريات الحياة ودخول عصر الحاسبات الالكية أصبح وجود نظام آلي يختص بالقرآن الكريم أمراً مفيداً للغاية ، حيث يمكن الإستفادة منه في التعليم ، وفي القيام بالبحوث المختصة بالقرآن الكريم ، وللاستعمال الخاص بالأفراد .

والهدف الرئيسي من هذا البحث هو تطوير نظام لعرض وتخزين واسترجاع القرآن الكريم ويقوم النظام بعرض وطباعة النص القرآني بالرسم العثماني . ويمكن البحث عن كلمة ، أو مقطع من آية ، أو حزب ، أو جزء ، أو آية ، أو سورة في القرآن الكريم . بالإضافة إلى أنه يقوم بعرض الأحكام المتعلقة بالتجويد في آية آية في القرآن الكريم . ويتميز النظام بسهولة التطوير والتغيير فيه . والنظام عربي بالكامل ويستخدم قوائم اختيار الوظائف ويعمل على أجهزة أي بي إم الشخصية أو المطابقة لها وهي الأكثر شيوعاً بين الناس .

درجة الماجستير في العلوم
جامعة الملك فهد للبترول والمعادن
الظهران - المملكة العربية السعودية
يونيه ١٩٨٨ م

Chapter I

INTRODUCTION

Since the shine of Islam, Muslims have been taking care of their Holy Book, the Quran. Today, as we live in the computer and information age, a computerized system for the Holy Quran is of great use. This system can be utilized in education, research concerned with the Holy Quran, and personal use.

The objective of this research is to design and implement a retrieval and display system for the Holy Quran. The system displays and prints the Holy Quran text using Al-Rasm Al-Othmani style. It searches for a Sora, Aiah, Hezb, Juz, word, or phrase. In addition, it analyzes any Aiah for Tajweed rules. It is a modular and flexible system, i.e., efficient for additions and modifications. Furthermore, it is menu-driven and fully Arabic. The system is implemented on the IBM Personal Computers (PC) and the compatibles which are the most popular machines.

The thesis is divided into four parts : first, the introduction, which is this section, second, a survey of the

existing systems, which covers the current systems in this field, third, the design and implementation of the system, which explains in detail the overall design of the system and how every feature of the system is implemented, and last, the conclusion.

Chapter II

EXISTING SYSTEMS

Currently, few systems involving the Holy Quran have been developed. In this section, some of these system are discussed. In addition, its features, advantages and shortcomings are explained.

2.1 Salsabeel

This system [1] was developed by Applied Microsystems Technology Limited, London, United Kingdom. It provides the user with the following:

- The ability to get the number of occurrences of a letter, word in a part or the whole of the Holy Quran.
- A range of the Holy Quran can be specified. For example, the operations can be limited to a specific Sora of the Holy Quran.
- The display of an Aiah, Sora or more.
- The ability to print the text.

In addition, the system runs on the IBM - Personal Computers and the compatibles. It divides the screen into three parts, one for entering commands, another for the

definition for the function keys, and the last for displaying text. The first two parts are displayed in English and the third in Arabic.

The system has the following shortcomings :

- The lack of Harakat and Tajweed characters (accentuation marks).
- Every Aiah is displayed on a separate line.
- Tajweed rules are not shown.
- The system does not have a good display of Al-Rasm Al-Othmani style.
- The user needs to know English to use the system.
- Aiah numbers are displayed beside the text using English.

2.2 Al-Raed

This system [2] was developed by National Group Company (NGC). It consists of three separate products developed at different times. The first provides the search for a specific Aiah number in a specific Sora. The second provides the search according to the subject. Finally, the third product searches for any characters and this product can be applied to any text not only for the Quran. These three products provide the user with the following :

- The display of Aiah, Sora or a multiple of them.

- The search for words, or multiples of them, according to a subject, or for specific Aiah's.
- The ability to print text.

This system runs on Al-Raed computers only. It is fully Arabic and has Harakat (accentuation marks) on the text. The implementation was done using BASIC as a programming language.

The following remarks can be made about the system :

- Tajweed characters and rules are not shown.
- The display does not conform to Al-Rasm Al-Othmani.
- Can only be used on AL-RAED computers.
- Search does not give correct results sometimes.
- The need for the above three products to have all the functions.

2.3 Al-Alamiah

This system [3] was developed by Al-Alamiah for their machines Sakhr MSX computers. This system provides the following :

- The ability to search for a letter, word or multiple of them.
- It conforms to Al-Rasm Al-Othmani quality of displaying text (the existence of Harakat and Tajweed characters).
- Information about the occurrences of a letter or a

word.

- The search according to the root of the word.
- Information about the meanings of the words.
- The ability to print text .
- A fully Arabic package.

A data compression technique, which is used for storing the Holy Quran text, characterizes this system in its functionality and usefulness in saving space. It is based on storing the root of the word only. Whenever a word is to be displayed, some characters are added to the root word and the corresponding Harakat and Tajweed characters are augmented to the word. Assembly language of Sakhr MSX is used to implement the system. However, the following remarks are noticed about the system :

- The process of displaying text is slow. This is due to the calculation of word lengths compared to the width of the screen, since words can not be split.
- Every screen displays three lines of text only.
- Can run on Sakhr MSX systems only.
- The system is not portable, since it is implemented in assembly language.
- Tajweed rules are not shown.

As this survey shows, Salsabeel and Al-Raed systems are not acceptable because they do not provide a good display of Al-Rasm Al-Othmani style and do not display Tajweed characters with the Holy Quran text. Al-Alamiha system has many advantages; however, it is restricted to Sakhr MSX computers. In addition, none of these systems provides Tajweed rules analysis. This research work is developed to overcome the shortcomings of the systems surveyed.

Chapter III

DESIGN AND IMPLEMENTATION

The main objective of this research is to design and implement a retrieval and display system for the Holy Quran. The system has the following specifications :

1. The display of the Holy Quran text is a good Al-Rasm Al-Othmani style similar to the Holy Quran available on paper. This means that Harakat and Tajweed characters (accentuation marks) are included in the system.
2. A multi-index search for a Sora, Aiah, Juz, Hezb, word, or phrase is available in the system.
3. The analysis for Tajweed rules is provided as a new feature. These rules are useful in education. The user is able to get the rules to aid him pronouncing the words in the proper way.
4. The ability to display and print the required text.
5. A fully Arabic package. All menus and messages are displayed in Arabic.
6. The system runs on the IBM-PC's and the compatibles. This is a good feature for the users because these PC's are the most popular systems.

In this chapter, a detailed description of the overall design and implementation of the system and its components are discussed. The system consists of six components :

1. User Interface Module,
2. Graphics and Contextual Analysis Module,
3. Storage Module,
4. Data Entry Module,
5. Search Module, and
6. Tajweed Rules Module.

It is decided to implement a menu-driven system. This means that the user has the facility to choose one of the options displayed on the screen. According to the choice another screen of information (or another screen of choices) will be displayed, and so on until the required information is obtained. Menu-driven systems are simple to use and friendly with the user. In addition, the system is designed to be modular. The modifications and improvements of the system can be done with flexibility and ease. The programming language used (TURBO PASCAL) has effects on some of the design issues of the system.

Figure 1, shows the block diagram of these components and the interconnections between Modules. It clarifies the

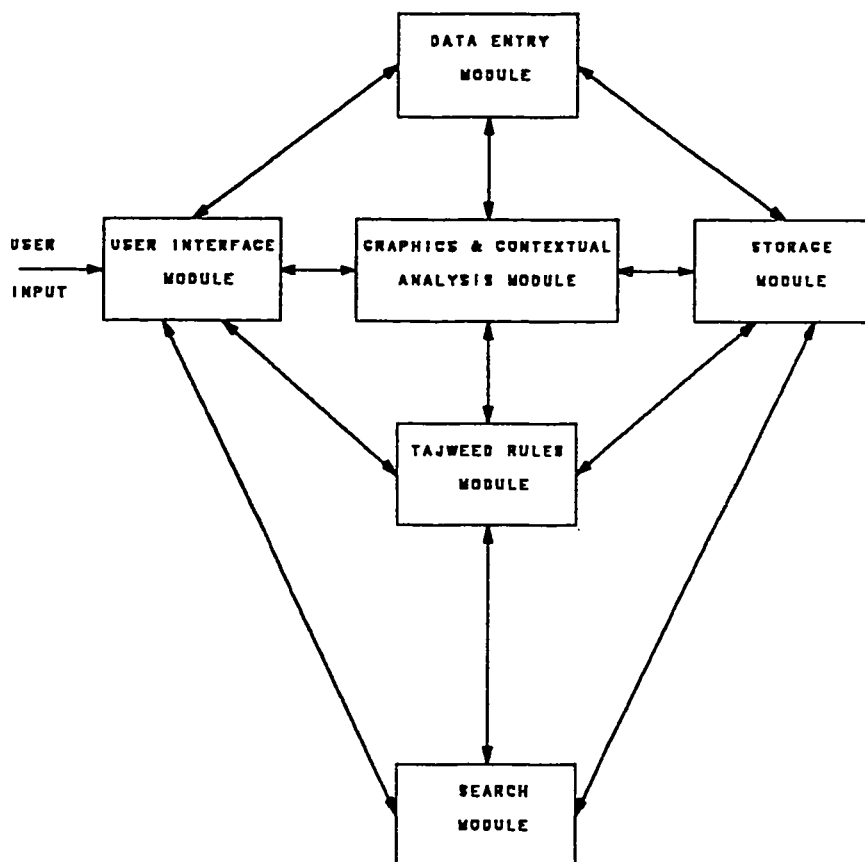


Figure 1 Block Diagram Of The System

modularity of the system. If a new feature or operation is to be added, a new block with the interconnections to the other components are added.

The user interface module is the interface between the user and the system. It interacts directly with the user by accepting his requests and then activates the corresponding module to generate the output.

The graphics and contextual analysis module is used to display menus, information, messages, and the Holy Quran text all in Arabic.

The storage module specifies the organization of the Holy Quran's data stored. The performance of the system depends on the design of this module and the way it stores data.

The data entry module implements the data entry process of the Holy Quran text. A simple editor is implemented to aid the data entry personnel in entering data.

The search module handles the operation of the multi-index search. It consists of few components that are activated from the search menu which is activated from the main menu.

The Tajweed rules module scans an Aiah specified by the

user for applications of Tajweed rules . The module shows the basic rules and gives the definitions of these rules if needed. The main menu activates this module.

3.1 User Interface Module

The function of this module is to display the menus, accept the user's option, call the proper module to perform the operation that the user chooses, and generate the output back to the user.

This module produces all the menus displayed by the system (the main menu and the sub-menus). The menus require a user input of a choice or specific data (like Sora name) to perform the operation. The menus, messages, choices, and information are displayed in Arabic.

This module has the ease, simplicity, and flexibility as major advantages. It interacts with the data entry, graphics and contextual analysis, Tajweed rules, and search modules.

The module simply displays menus of choices and waits for the user input. A small procedure checks the validity of the input. Then, the module activates the corresponding module or procedure to handle the operation required by the user. The module uses the routines of the graphics and contextual analysis module to display these menus.

3.2 Graphics And Contextual Analysis Module

This module performs the following operations :

- a. analysis and display of Arabic text with Harakat and Tajweed characters (accentuation marks),
- b. Analysis and display of Arabic text without Harakat and Tajweed characters (accentuation marks), and
- c. generation of any shape for letters, Harakat and Tajweed characters, numbers, or for any other purposes (not necessarily Arabic).

The module contains two separate parts, the graphics and the contextual analysis.

3.2.1 Graphics Module

All of the shapes used by the system are generated using the graphics module. In addition, the module draws these shapes whenever needed.

A separate routine is used to aid in the process of generating the shapes. This routine produces another routine that draws the shape generated. The following section describes how it works.

3.2.1.1 Shape Generation

The routine provides the ability to generate a new shape or to modify an old one to produce the required shape. This routine displays a menu of three options :

1. generate a new shape,
2. modify an old shape, and
3. exit the routine.

Regardless of the option chosen (except the exit), a matrix of 10 X 20 positions is displayed. The shape to be generated can be different in size. Actually, the shapes generated are variable length (size). For time savings, shapes already generated can be used and modified to produce new shapes.

3.2.1.1.1 New Shape Generation If a new shape is to be generated, a matrix of 10 X 20 positions is displayed. These dimensions are the maximum needed to generate the shape.

The routine prompts for the vertical and horizontal positions. The corresponding rectangle on the matrix is filled accordingly. If the same vertical and horizontal positions are entered again, the rectangle is blanked.

The procedure is repeated again until the required shape is obtained. Then, the shape can be stored as a routine which draws the shape again. Otherwise, the module ignores the shape generated. If stored, the file and routine names are prompted to be identified for later use. The file used is a text file that contains the code of the routine. The code of the routine contains statements that move the cursor and plot a dot in specified positions using the graphics capabilities of the programming language.

3.2.1.1.2 Old Shape Modification This option can be utilized to enhance the shape. In addition, similar shapes can be generated using old stored ones to save time.

The file and routine names are prompted for the old shape. In case that the file or routine names are not found, an error message appears and the names are prompted again. Then, the matrix filled with the shape is displayed. The vertical and horizontal positions are prompted and the procedure in the previous section is repeated.

3.2.1.2 Shape Drawing

This part is used to draw the specified shape by activating the corresponding routine generated using the shape generation routine mentioned in the previous section.

The shapes consists of :

- a. letters,
- b. Harakat and Tajweed characters,
- c. numbers, and
- d. special characters (e.g., '(').

Since the writing in Arabic requires the letters to be connected with each other, four shapes for each letter are needed. The four shapes are :

- a. begin-of-word shape,
- b. middle-of-word shape,
- c. end-of-word shape, and
- d. isolated letter shape.

For the rest of the characters (non-letters), only one shape is needed. The contextual analysis module determines the shape needed for drawing. The routine uses codes for each character and for the shape to draw the right shape. Table(1) shows these codes (ASCII codes), and the corresponding Arabic and English

HEX	ENG	ARB	HEX	ENG	ARB	HEX	ENG	ARB	HEX	ENG	ARB
01	^A	ا	37	7	٧	4F	O	و	67	g	ج
02	^B	ب	38	8	٨	50	P	ز	68	h	و
05	^E	هـ	39	9	٩	51	Q	ر	69	i	ي
21	!	!	3A	:	:	52	R	ز	6A	j	ي
22	"	و	3B	;	;	53	S	س	6B	k	ك
23	#	ع	3C	<	<	54	T	ث	6C	l	ل
24	\$	د	3D	=	=	55	U	ص	6D	m	م
25	%	ف	3E	>	>	56	V	ض	6E	n	ن
26	&	ع	3F	?	?	57	W	ط	6F	o	و
27	'	'	40	@	@	58	X	ظ	70	p	پ
28	()	41	A	أ	59	Y	ع	71	q	ق
29)	(42	B	ب	5A	Z	غ	72	r	ر
2A	*	* ¼	43	C	ج	5B	[ق	73	s	س
2B	+	* ½	44	D	د	5C	\		74	t	ت
2C	,	* ¾	45	E	هـ	5D		ا	75	u	و
2D	-	-	46	F	ف	5E	^	ا	76	v	و
2E	.	.	47	G	ج	5F	_	ا	77	w	و
30	0	٠	48	H	هـ	60	`	ا	78	x	خ
31	1	١	49	I	ة	61	a	ا	79	y	ي
32	2	٢	4A	J	ت	62	b	ب	7A	z	ز
33	3	٣	4B	K	ث	63	c	ك	7B	{	ك
34	4	٤	4C	L	ج	64	d	ل	7C		ل
35	5	٥	4D	M	ح	65	e	م	7D	}	م
36	6	٦	4E	N	خ	66	f	ن	7E	~	* ¾

Table 1 ASCII TABLE

equivalents.

3.2.2 Contextual Analysis Module

The function of this module is to analyze the text to be drawn to determine the correct shapes for the characters [4,5], and call the shape drawing routine to draw the text itself.

Letters are the only characters that need to be analyzed, because they have four shapes for each. The letters are classified into four classes :

1. Non-connectable letters,
2. Letters connected from the right only,
3. Letters connected from the left only, and
4. Letters connected from both sides.

The module contains two sub-modules, a contextual analysis for reading data that has been displayed on the screen, and a contextual analysis that determines the correct shape and call the corresponding routine to draw shape. The main reason for that is to enhance the performance of the display timewise. The routine used for reading erases the previous character drawn and draws it again if new ones are entered, because the shape of the character erased has been changed. If this routine is used

to draw data on the screen, each time a character is drawn, it is erased and a new shape is drawn. Both of the routines are tested practically and a great enhancement in time is found using the second routine for data display.

3.2.2.1 Contextual Analysis For Data Reading

This module is responsible for drawing Arabic text on the screen and read it for later manipulation. It is used mainly by the data entry module. Other modules use it for reading data displayed on the screen (for example, Sora name).

The module is divided into routines that organize the analysis operation. These routines are :

- a. class determination,
- b. shape size determination, and
- c. shape erase.

The module uses the class-determination routine to determine the class of the character mentioned above. The number of pixels needed to draw the character is obtained using the shape size determination routine. The cursor is moved according to the character size obtained previously. When a character is to be erased, the shape-erase routine is used.

The block diagram of the module is shown in Figure 2.

3.2.2.1.1 Class Determination This routine gets a character as a parameter and produces an integer which is the class of the character. A simple check on the character (using a CASE statement) specifies the character class.

3.2.2.1.2 Shape Size Determination The shapes generated have different sizes to give a better display of the text. Each letter has a different size depending on its class. Therefore, there is a need to determine the size of the shape (number of pixels) to specify the movement of the cursor on the screen.

A routine is implemented to perform this job. By having the character and its class, passed as a parameter, it returns an integer number indicating the size of the character. It is a simple routine which checks the character and the class (using IF statements) to give the size.

3.2.2.1.3 Shape Erase This routine is used to erase the character previously displayed. The purpose of that is to rewrite the character with a

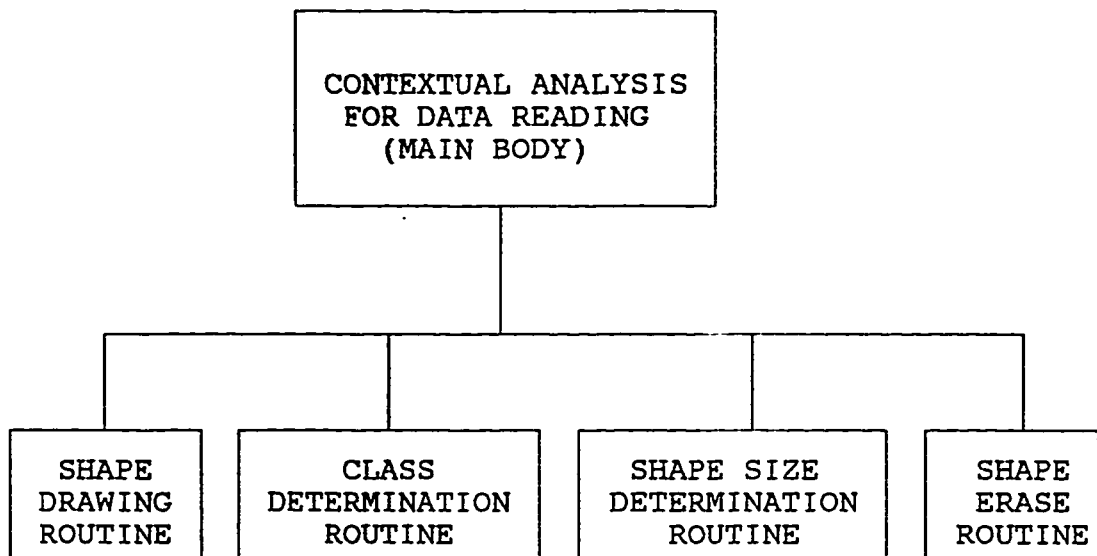


Figure 2 BLOCK DIAGRAM OF THE CONTEXTUAL ANALYSIS
FOR DATA READING ROUTINE

different shape after determining the correct class of the next character. This is needed for letters only. The routine simply blanks the previous position specified by a parameter (it is the size of the character obtained using the shape size determination routine).

The contextual analysis for data reading module works in a character-by-character fashion. It draws one character and the corresponding accentuation mark (passed as parameters) each time it is called. If one of the parameters (or both) is blank (space) then it is not drawn and the position is left blank.

This module performs its operation in the following manner :

- if the parameter can not be drawn, then exit the module,
- get the class of the character by calling the class determination routine,
- if the character can not be connected with the previous one (class 1 or 3) , then draw the character on the next position,
- if it is can be connected with the previous one, (class 2 or 4) then :
 - begin1

- if the previous letter can be connected with the following one (class 3 or 4), then :
 - begin2
 - get the size of the previous letter by calling the shape size determination routine,
 - erase it by calling the shape erase routine,
 - set the shape of the character to be the middle-of-letter shape or the begin-of-letter shape depending on the shape of the previous letter.
 - draw the previous letter using the new class,
 - draw the corresponding accentuation mark,
 - move the cursor using the size of the character just drawn,
 - check for a special case for the Arabic letter (ﻯ) and adjust the variable accordingly for the current letter,
 - end2.
- if the previous letter can not be connected with the following one (not class 3 or 4) then set the current letter to be not connected with the previous one,

- draw the letter and the accentuation mark using the current class set previously,
- adjust the global variable for the next operation for the special case mentioned previously,
- endl.
- move the cursor using the size of the current character, and
- store the current character and the accentuation mark to be used later.

The routine draws the end-of-letter shape for the last character in the text. Then, it adjusts it if there is a need.

3.2.2.2 Data Drawing Contextual Analysis

The function of this module is to draw text in Arabic on the screen. It is mainly used to display the Holy Quran text, messages, menus, and other information. The previous module can be used for the same purpose, but it is slow compared to this module. The reason is that the previous module erases the letters displayed on the screen and draws them again with a different shape to reflect the existence of new characters in the word. Both of these modules are used

practically and this one found to be much more efficient in time.

The module draws word-by-word without erasing any previously drawn character. It draws the word and the corresponding accentuation mark (passed as parameters) each time it is called. A boolean variable passed as a parameter determines if the accentuation marks are to be drawn. An array which saves the values of the shapes of the characters to be drawn is used to perform the drawing.

The block diagram of the module is shown in Figure 3.

The module performs its operation in the following manner :

- set the class of the previous letter to be non-connectable and the shape variable to be isolated shape,
- if the length of the word is greater than 1 (not a letter) then :
 - begin1
 - loop1
 - get the class of the character by calling the class determination routine,
 - if the character can not be connected with the

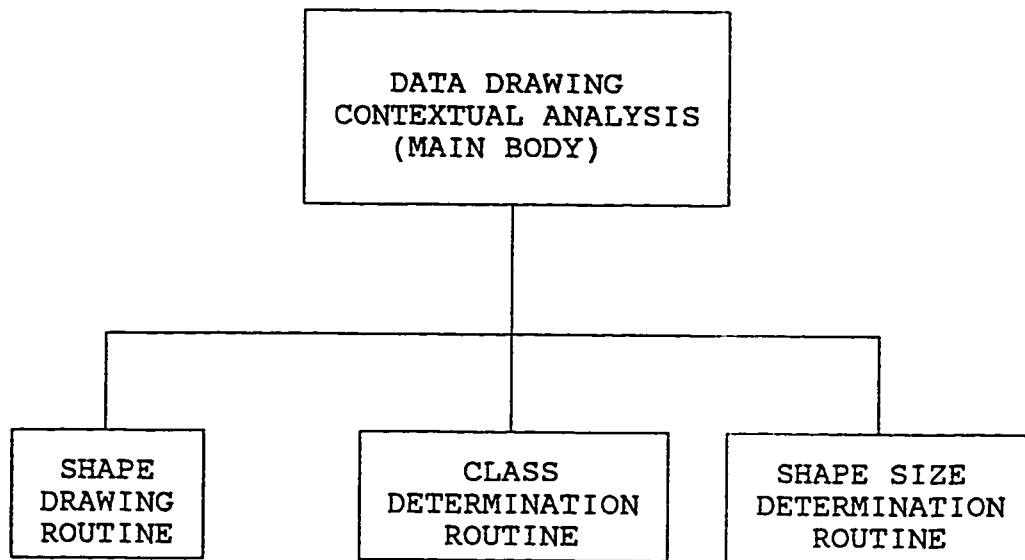


Figure 3 BLOCK DIAGRAM OF THE DATA DRAWING
CONTEXTUAL ANALYSIS ROUTINE

previous one (class 1 or 3), then save the shape variable in the array and set it to be of isolated shape,

- if it can be connected with the previous one (class 2 or 4), then :
 - begin2
 - if the previous letter can be connected with the following one (class 3 or 4), then :
 - begin3
 - set the shape of the current character to be the middle-ofletter shape or the begin-of-letter shape depending on the previous shape of the character. Then store the value in the array,
 - check for a special case for the Arabic letter (ﻯ) and adjust the shape accordingly for the current letter,
 - end3
 - if the previous letter can not be connected with the following one (not class 3 or 4) then set the current letter to be not connected with the previous one,
 - end2
- save the current shape variable in the array and

```
    save the class of the character,  
- repeat the previous operations for the rest of the  
  characters of the word,  
- endloop1  
- if the boolean variable indicates to draw the  
  accentuation marks then :  
  - begin4  
  - for the length of the word do :  
    - begin 5  
    - draw the letter and its accentuation mark,  
    - obtain the character size by calling the shape  
      size determination routine mentioned in the  
      previous module,  
    - move the cursor using the size of the character  
      just drawn,  
    - end5.  
  - end4.  
- else  
  - begin6  
  - for the length of the word do :  
    - begin7  
    - draw the letter ,  
    - obtain the character size by calling the shape  
      size determination routine mentioned in the
```

```
        previous module,  
        - move the cursor using the size of the character  
          just drawn,  
        - end7  
    - end6.  
- endl.  
- else (if the word is a letter) then :  
    - begin8  
    - draw the letter and the accentuation mark (if  
      available),  
    - obtain the character size by calling the shape size  
      determination routine mentioned in the previous module,  
    - move the cursor using the size of the character just  
      drawn,  
    - end8.
```

3.2.3 Other Related Routines

There are other routines that are related to the graphics and contextual analysis module. They are used by more than one module. These routines are :

- a. draw a number,
- b. read a number,
- c. draw a phrase,
- d. draw a menu border,

- e. display appendix, and
- f. text printing.

3.2.3.1 Number Drawing

The function of this routine is to draw a number passed as a parameter. It is used to draw Aiah numbers and the numbers on the menus.

The routine performs the following operations :

- if the number is greater than 9 (more than one character to be drawn) then :
 - the number is divided by 10,
 - the remainder of the division is saved in a variable and the result of the division is saved using the same variable passed as a number, and
 - the routine is called recursively for both of the values.
- else (the number is less than 10) then :
 - the number is drawn and the cursor is moved 6 places (the length of the number).

3.2.3.2 Number Reading

The function of this routine is to draw Arabic numbers on the screen, read the codes from the screen,

and convert it to an integer number. The routine is called to read numbers entered by the user (like Juz number). The numbers are entered in a reverse order (the Arabic way). For example, if 367 is to be entered, the user should press 7, 6, then 3.

The routine performs the following operations :

- displays a cursor to indicate that it is waiting for an input,
- reads the input,
- if the input is a number, then :
 - begin1
 - the number is added to a variable,
 - if the number exceeds the limit an error message is displayed and the cursor does not move to indicate that the input should be modified,
 - else the number is drawn,
 - endl.
- if the input is a back space then :
 - the last number drawn is deleted from the variable and it is erased from the screen,
- if the input is a carriage return (CR) then:
 - begin2
 - if the input is blank then an error message is

displayed and it waits for the input again,
- else the variable that holds the number is
converted to an integer and the routine
terminates,
- end2.

3.2.3.3 Phrase Display

The function of this routine is to display any text (passed as a parameter) of more than one word of length and less than one line of text. This routine is used to display messages, menus, and information. It calls the contextual analysis routine that is responsible for the data display. The routine is implemented to save code, because there is no need to perform the same operations each time a message is to be displayed.

The routine performs the following operations :

- for the length of the text :
- each word is obtained separately and the contextual analysis module is called to display it.

3.2.3.4 Menu Border Drawing

The function of this routine is to draw a boarder on the screen to give a good looking of the screen. The

color of the border is passed as a parameter to the routine.

The routine performs the following operations :

- draws four lines at the edges of the screen using the color specified.

3.2.3.5 Appendix Display

The function of this routine is to display an appendix showing information about the Sora's of the Holy Quran. The routine displays a table which contains the Sora number, Sora name, Juz number(s), Hezb number(s), and number of Aiah's in Sora. The routine displays messages of information to perform its operation. It calls the contextual analysis routine that is responsible for the data display. The routine displays one screen of information and prompts for an input. The user has the ability to continue the display of the appendix by a key press of the (PgDn) key or terminate the process by pressing the (CR) key.

3.2.3.6 Text Printing

There is no actual routine implemented for printing the text. The system uses the utilities in the DOS operating system to print text. The print screen

utility is used to print text. A routine (GRAPHICS.COM) has to be activated prior to using the print screen utility.

3.3 Storage Module

This module specifies and determines the data organization of the Holy Quran. Of the two design choices considered, one is justified and implemented. The module interacts with the data entry module to accept the data of the Holy Quran.

The following points explain the design issues :

- a. The Holy Quran is divided into thirty parts (Juz) as is on papers,
- b. Each part is stored into two files, one for text (alphabetical characters of the Arabic language), and the other for accentuation marks (Harakat and Tajweed characters), and
- c. Each file consists of records of string characters.

Two choices were considered in the design. The more efficient one in terms of space and time combination is implemented. The two choices are, to use random access or text files. The programming language used has some restrictions on files. It does not provide variable size records. Therefore, variable size records were not

considered in the design.

3.3.1 Random Access Files

This choice has the following points:

- a. The record length has a specific fixed size (e.g. 255 characters),
- b. The record is filled of data (no blanks) to save disk space. This means that a record may contain more than one Aiah,
- d. A table is established showing the place of each Aiah in the Holy Quran (which record, position in the record, and file), and
- e. Other tables are established for the search process.

3.3.2 Text Files

The following points are considered in this design choice :

- a. An end-of-Aiah character is concatenated at the end of each Aiah (indicates if an Aiah occupies more than one record (line)),
- b. Each Aiah, is stored in one line unless its length is greater than 255 characters (the limitation of the strings of the programming language),
- c. If the length of an Aiah is greater than 255

characters, the first 255 characters are stored in the first line, the next ones in the next line, and so on until all the characters are stored (the last line contains the end-of-Aiah character) (the longest Aiah in the Holy Quran occupies four lines),

- d. Tables are needed for the search process to enhance the speed of the process by retrieving the required data directly from the files using the information in these tables.

3.3.3 Comparison Of The Choices

The choices are compared in terms of space and time. The search process is the major factor of the speed. The search options are :

- a. specific Sora,
- b. specific Aiah in a Sora,
- c. specific Juz,
- d. specific Hezb, and
- e. specific word or phrase.

3.3.3.1 Search For A Specific Sora

Random access files are faster, since the tables direct to the specific record in the file and using the seek function the record is reached. In case of the

text files, the table of Sora determines the position of the Sora in the file but a (Readln) is needed starting at the top of the file until reaching the specific position.

3.3.3.2 Search For A Specific Aiah In A Sora

For the same reason above, random access files are faster.

3.3.3.3 Search For A Juz

The two options need the same time, since each Juz begins at the beginning of the file.

3.3.3.4 Search For A Hezb

Random access files are faster for the same reason mentioned in 3.3.3.1.

3.3.3.5 Search For A Word Or Phrase

If the word is found in the word table, random access files retrieve faster for the same reason in 3.3.3.1. If it is not found in the word table, both options take the same time, since the search is sequential.

Another factor that affects the speed is the display of the Holy Quran.

3.3.3.6 Display Speed

Random access files option needs to calculate the length of the Aiah to be displayed using the table of Aiah information. Then the Aiah is displayed. On the other hand, text files need to check for the end-of-Aiah character to determine Aiah boundaries. This means that text files are faster because there is no need to retrieve data from the table and no calculations are involved.

3.3.3.7 Space Consideration

Random access files option needs

6226 (no. of Aiah in the Holy Quran)

4 x (no. of bytes for each entry in the
table)

24904 bytes

for the Aiah table. for the other tables, both options require the same space.

3.3.3.8 Conclusion

It is clear that the space requirements for random access files limits its applicability as an efficient design choice. In addition, it needs more time for the display process. The advantage is in the search process, but it is found out that text files do not degrade or slows the search process significantly if compared to random access files.

Therefore, the option of text files was chosen and implemented.

3.3.4 Advantages Of The Design

The design of the module has the following advantage :

- a. Faster sequential search since the characters are stored in one file and accentuation marks in another one (only characters file is needed for the search),
- b. Ease in the implementation of the module (no complications in terms of tables, data structure, etc.), and
- c. It is a compromise in terms of space and time, using tables to index the needed information.

3.4 Data Entry Module

The objective of this module is to aid in the data entry process (data storage). A simple basic editor is used in this module. The module interacts with the graphics and contextual analysis module to display the data stored. In addition, it interacts with the storage module to organize the data as specified there.

This module has three options, store or add data to a specific Juz, delete a specific Juz, and modify a specific word in a specific line in a Juz.

The block diagram is depicted on Figure 4.

The module has five parts :

- a. the main procedure,
- b. file determination,
- c. add data to a Juz,
- d. delete a Juz, and
- e. modification of a Juz.

3.4.1 The Main Procedure

The procedure displays a menu which has three option, add data, delete data, modify data, and exit the module.

According to the input, this procedure invokes the corresponding part of the module unless the input is to exit

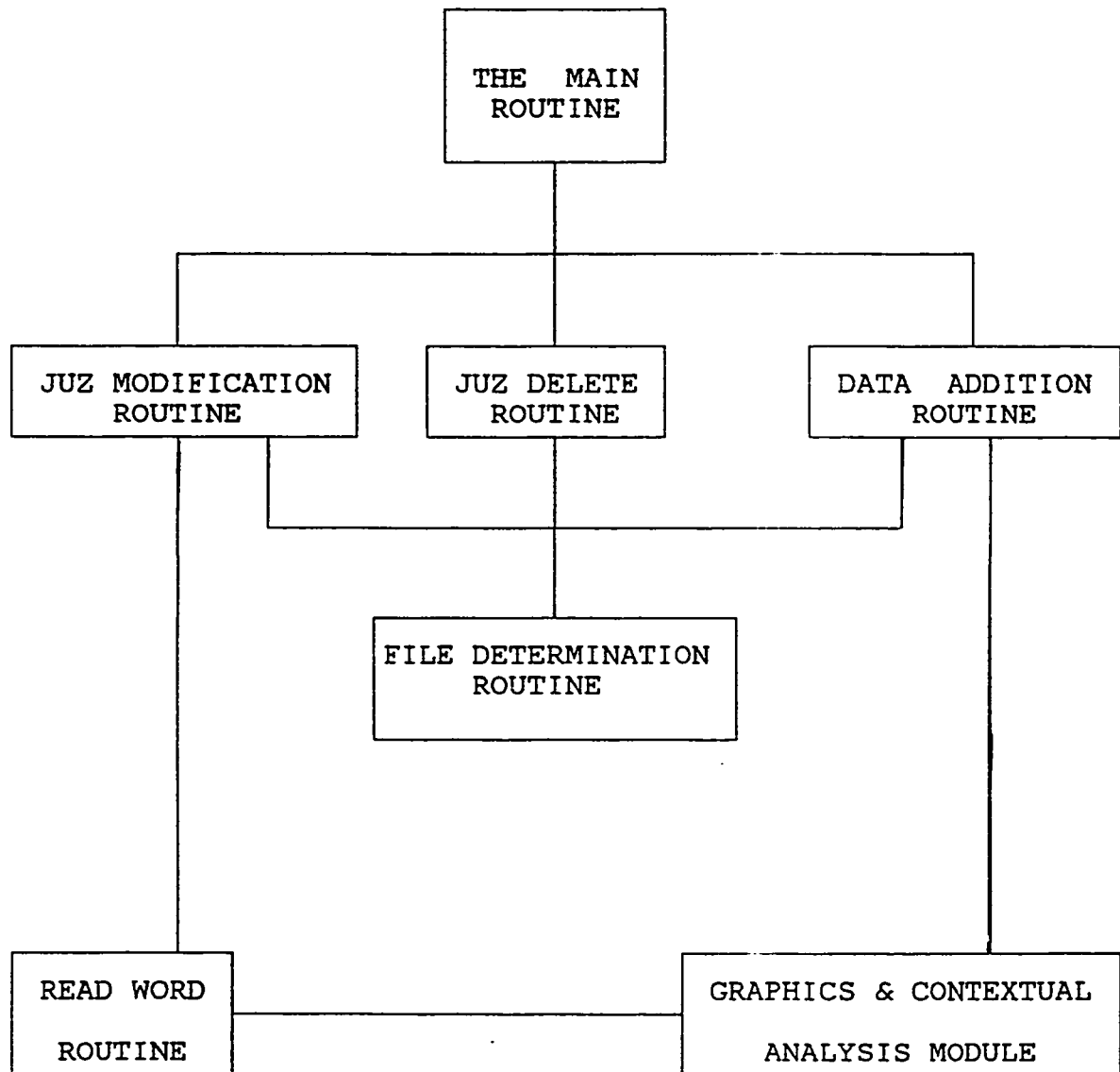


Fig 11 BLOCK DIAGRAM OF THE DATA ENTRY ROUTINE

the module.

3.4.2 File Determination

This routine accepts a number as an input from the screen, manipulates the number to reflect a file name, determines the status of the file, and returns these information to the calling procedure.

It performs the following operations :

- displays a message on the screen asking the user to input a Juz number,
- reads the number using the (get-num) procedure of the contextual analysis module,
- concatenates a specific word to the number to produce a file name for the character file,
- checks for the existence of the file and stores the result in a flag,
- produces a file name for accentuation mark file, and
- prints a message if the file is new in case that the procedure is called by the data addition part.

3.4.3 Data Addition (Storage)

This routine adds data to a Juz using a simple editor. It stores data according to the storage module organization. In addition, it interacts directly with the contextual

analysis module.

The following is performed by this routine :

- calls the file determination procedure,
- if the files exist, the files are positioned at the end of the previous information stored using the (APPEND) function, otherwise, new files are created,
- a new screen is displayed and a cursor appears showing a ready mode for data entry,
- a loop starts to store more than an Aiah,
- two strings for characters and accentuation marks are blanked,
- the editor starts accepting input from the user,
- if the input is in the set of the Arabic alphabetical characters;
 - begin1
 - the accentuation mark is read (the cursor disappears to show prompt for the mark),
 - if the character read is not an accentuation mark, then read it again,
 - the contextual analysis module is called (the character-by-character part) to draw the input,
 - the input is added to the strings,
 - arrays of the classes of the character, the previous

- character, and its type are updated to reflect the input (these arrays are used by the editor to know the previous information of the characters),
- if the input is greater than 255, then the strings are written in the files by (WRITELN), the strings are blanked, and the array index is set to zero,
 - endl.
 - if the input is a carriage return (CR), then the strings are stored in the files by (WRITELN) after concatenating them with the end-of-Aiah character, and the editor is exited.
 - if the input is a back-space then
 - begin2
 - if the array index is zero (there is no more characters in the strings), then nothing is done,
 - the length of the character is determined and it is erased,
 - the character and accentuation mark are removed from the strings and the array index is decremented,
 - if this index is zero, then the parameters of the contextual analysis are set,
 - otherwise, the previous character, accentuation mark, and the character type are updated using the arrays,

- the previous character is erased and drawn again as an endof-letter shape and,
- the arrays are updated for this change,
- end2.
- if the display is about to reach the end of the line, the display is done on the next line if the end of the screen is not reached, otherwise the screen is blanked,
- if the editor is left, a message is displayed giving the user the option to leave the date entry screen (loop) or continue for the next Aiah, and
- at the end, the files are closed and the routine returns to the main procedure.

3.4.4 Deletion Of A Juz

This routine deletes a Juz file if the user needs that. It performs the following :

- calls the file determination procedure,
- checks for the existence of these files,
- if they are not found, a message indicating that is displayed and exit the routine, and
- otherwise the files are erased and a message is displayed reflecting the operation.

3.4.5 Modification Of A Juz

This routine deletes text and its accentuation marks stored earlier and replaces them by new text. The text removed and replaced is determined by the user. The routine uses another one that displays and reads the old and new text. The routine saves the old files (character and accentuation marks files) and creates new ones with the changes added to them. The routine interacts with the graphics and contextual analysis module.

3.4.5.1 Read Text

This routine displays text entered by the user, reads the input, and returns the text to the calling routine.

The routine performs the following operations :

- two strings for characters and accentuation marks are blanked,
- loop1
 - the routine starts accepting input from the user,
 - if the input is in the set of the Arabic alphabetical characters, then
 - begin1
 - the accentuation mark is read (the cursor disappears to show prompt for the mark),

- if the character read is not an accentuation mark, then read it again,
- the contextual analysis module is called (the character-by-character part) to draw the input,
- the input is added to the strings,
- arrays of the classes of the character, the previous character, and its type are updated to reflect the input (these arrays are used by the editor to know the previous information of the characters),
- if the input is greater than twenty, then quit the loop and accordingly exit the routine,
- endl.
- if the input is a carriage return (CR), then exit the routine.
- if the input is a back-space then
 - begin2
 - if the array index is zero (there is no more characters in the strings), then nothing is done,
 - the length of the character is determined and it is erased,
 - the character and accentuation mark are removed from the strings and the array index is decremented,

- if this index is zero, then the parameters of the contextual analysis are set,
- otherwise, the previous character, accentuation mark, and the character type are updated using the arrays,
- the previous character is erased and drawn again as an end-of-letter shape and,
- the arrays are updated for this change,
- end2.
- endloop1.

The modification-of-a-Juz routine performs the following operations :

- calls the file determination procedure,
- checks for the existence of these files,
- if they are not found, a message accordingly is displayed and the routine is exited, and
- displays a message asking the user to enter the line number,
- calls the get-number routine that reads the number,
- displays a message asking the user to enter the text to be removed,
- calls the read-text routine to read the text,
- displays a message asking the user to enter the text to be replaced,

- calls the read-text routine to read the text,
- renames the files as backup files,
- opens new files,
- loop1 (for the line number entered by the user),
 - reads a line from the backup files and stores it in the new files,
 - if the end-of-file is reached then display a message indicating an error in the line number and exit the routine,
 - endloop1
- reads the current line and find the text to be removed in the line,
- if it is not found in the character and accentuation marks then display a message accordingly and exit the routine,
- if it is found then delete the text from the line and replace it with the new one (using built-in functions),
- if it is found in one of the character or the accentuation marks, it displays a message to the user asking him to enter more text to be identified and exit the routine,
- for the rest of the lines remaining in the backup files, copy them into the new files,

- closes all of the files.

3.5 Search Module

The purpose of this module is to :

- a. search for a Sora, Aiah, Juz, Hezb, word, or phrase,
- b. display the text starting from the search index specified previously, and
- c. continue the display until user termination or the end of text.

The module interacts with the storage module to retrieve the required text. In addition, it interacts with the graphics and contextual analysis module to display the text and messages on the screen. It is activated by the user interface module upon user's request. Furthermore, it interacts with the Tajweed rules module to provide it with the required information about the Aiah needed for Tajweed rule analysis (searches for the Aiah).

This module provides the user with the following options :

- a. search for a specific Aiah,
- b. search for a specific Sora,
- c. search for a specific Juz,
- d. search for a specific Hezb,
- e. search for a specific word, and
- f. search for a specific phrase.

The module is divided into many parts (routines) to perform these operations. These routines are :

- a. the main routine,
- b. read text,
- c. initialize tables,
- d. Sora and Aiah search,
- e. Juz search,
- f. Hezb search,
- g. word and phrase search, and
- h. display text.

The main routine activates the other routines depending on the user's choice. Each routine prepares the information needed by the display routine. Then, the display text routine handles the rest of the operation. It continues to display text until the end of the Holy Quran text or user termination of the operation. For efficiency purposes, two operations are combined into one routine (Sora and Aiah search, and word and phrase search). Tables and data structure are used to enhance the execution of the module. They are described in the corresponding routines and shown in the figures defined later.

3.5.1 The Main Routine

The routine displays the search menu, checks for the input, and activates the corresponding routine. An exit from the module is provided as an option in addition to The option mentioned above. Upon user's request, the corresponding routine is activated unless the request is to exit the module.

3.5.2 Read Text

The routine reads a user's input of text. It is used to read data needed by the system (for example, Sora name). A simple basic editor is used again. It is similar to the one mentioned in the data entry module, but accentuation marks are not used.

The routine performs the following operations :

- prompts for an input by displaying a cursor,
- reads any key pressed by the user,
- if the key is a carriage return, then the routine finishes its function and returns back to the calling routine,
- if it is a back space then the last character is erased and the related variables are manipulated (for more details, refer to the data entry module),
- if the key pressed is not a carriage return or a back

space and it is allowable, then the corresponding character is displayed and the related variables are manipulated (refer to the data entry module for more details),

- the routine repeats the above operations until a carriage return (CR) is pressed or the length of the variable read reaches the maximum allowable.

3.5.3 Initialize Tables

The routine assigns values to the tables used by the search module to enhance the process. This routine consists of three other routines. These routines are :

- a. initialize Sora table,
- b. initialize Juz table,
- c. initialize Hezb table, and
- d. initialize word table.

These routines simply consist of assignment statements which assign the right values to the table variables. The structure of these tables are discussed in the following sections, each in the corresponding search routine.

3.5.4 Sora And Aiah Search

The routine prepares the information needed to display the text required by the user. Then, the display text

routine handles the rest of the operation. This routine performs two operations at the same time. It searches for a specific Sora and Aiah. These operations are combined into one routine, because of the similarity in the actions taken. A parameter is used to specify the option of performing a search for a Sora or for an Aiah.

A table (Sora table) of information about each Sora is established for this operation. Each Sora has an entry in the table. Therefore, the table size is 114 (number of Sora's in the Holy Quran). Each entry in the table is a record. The record contains the following data :

- a. Sora name (used to define the search index and to identify the Sora),
- b. number of Aiah's in the Sora (used for information display to the user and to aid in the search process),
- c. Juz number (identifies the storage file and used for display),
- d. start of Sora line number in the file (the exact location in the file for direct access),
- e. Sora classification (Makkiya or Madaniya) (used for information display to the user), and
- f. number of Aiah's of Sora in the file (used to indicate if Sora continues into the next Juz

(file)).

Figure 5 shows the structure of the table.

The routine performs the following operations :

- obtains the Sora name, by displaying a message prompting the user and then reading the name (by calling the read text routine),
- if the name read is not found in the Sora table, an error message is displayed, the name is erased from the screen, and it is read again,
- if found, then :
 - begin1
 - if the option is to search for an Aiah, then
 - begin2
 - another message prompting for the Aiah number is displayed and the number is read by calling the read number routine of the graphics and contextual analysis module,
 - if the number is bigger than the number of Aiah's in the Sora, an error message is displayed and the number is read again,
 - end2,
 - Juz number is obtained from the table,
 - if the option is to search for an Aiah and the Aiah required is greater than the number of Aiah's in the

SORA NAME	# OF AIAH	JUZ #	KIND	PLACE	AIAH IN JUZ
--------------	--------------	----------	------	-------	----------------

Figure 5 SORA TABLE

file (or Juz), then :

- begin3
- increment the Juz number by one,
- handle a special case for Sora number 2 and Sora number 4, since they are spread on more than two Juz,
- end3,
- augment the Juz number with a specified text to obtain file names for text and accentuation marks,
- check the existence of these files, display error messages if they do not occur, and exit the routine,
- position the file index at the beginning of the Sora (use READLN),
- if the option is to search for a Sora, then
 - begin4
 - set Aiah number to equal 1,
 - set the index that points to the line of the file to equal the start of the Sora,
 - obtain the Hezb number, by using the data in the Hezb table,
 - if the Hezb occurs in the same Juz that the Sora occurs in, then
 - if the start of the Sora is less than the start of the Hezb then increment Hezb number by one,

- end4,
- else (search for an Aiah)
 - begin5
 - search for the position of the Aiah,
 - read a line and check if it contains an Aiah,
 - if so, increment a variable that counts the number of Aiah's read and increment the index on the file,
 - else, continue reading until the end-of-aijah is reached (increment the index for each repetition),
 - continue the above operations until the Aiah needed is reached,
 - search for the Hezb number the same way explained above, (use the index here instead of the start of Sora value),
 - end5,
- call the text display routine to handle the rest of the operation,
- endl.

3.5.5 Juz Search

This routine prepares the information needed to display the text required by the user (begin of a Juz). Then, the

display text routine handles the rest of the operation.

A table (Juz table) is established to enhance the operation of the routine. Each Hezb has an entry in the table. The table size is 30. Each entry of the table is a record which contains the following data :

- a. Sora number (used for information display), and
- b. Aiah number (used for information display).

The rest of the data needed are obtained from the Hezb table.

Figure 6 shows the structure of the table.

The routine performs the following operations :

- obtain the Juz number, by displaying a message prompting the user for the number,
- read the number (by calling the get-a-number routine in the graphics and contextual analysis module),
- if the number is outside the range (1-30), then an error message is displayed and the number is read again
- augment the Juz number read with a specified text to obtain the file names for text and accentuation marks,
- check the existence of these files, display error messages if they do not exist, and exit the routine,
- set the file pointer to equal 1 (begining-of-file),
- obtain the Hezb number by getting (from Hezb table (shown later)):

SORA #	AIAH #
-----------	-----------

Figure 6 JUZ TABLE

- begin1
- the first Hezb number that occurs in the Juz is the required Hezb number,
- end1.
- obtain Sora number and the Aiah that reflect the start of the Juz (using Juz table),
- call the text display routine to handle the rest of the operation.

3.5.6 Hezb Search

This routine prepares the information needed to display the text required by the user (begin of a Hezb). Then, the display text routine handles the rest of the operation.

A table (Hezb table) is established to enhance the operation of the routine. Each Hezb has an entry in the table. The table size is 60. Each entry of the table is a record which contains the following data :

- a. Juz number (identifies the storage file and used for information display),
- b. start of Hezb in the file (the exact starting location used for direct access),
- c. Sora number (used for information display), and
- d. Aiah number (used for information display).

The rest of the data needed are obtained from the Sora

table.

Figure 7 shows the structure of the table.

The routine performs the following operations :

- obtain the Hezb number, by displaying a message prompting the user for the number,
- read the number (by calling the get-a-number routine in the graphics and contextual analysis module),
- if the number is outside the range (1-60), then an error message is displayed and the number is read again,
- the Juz number is obtained from the Hezb table,
- augment the Juz number with a specified text to obtain file names for text and accentuation marks,
- check the existence of these files, display error messages if they do not occur, and exit the routine,
- position the file pointer to the beginning-of-hezb using (READLN) operation (repeat the operation start-of-Hezb times (obtained from the table)),
- set the file index to equal the begin-of-hezb value obtained from the table,
- obtain Sora number and the Aiah that reflect the start of the Hezb from the table,
- call the text display routine to handle the rest of the operation.

JUZ #	PLACE	SORA #	AIAH #
----------	-------	-----------	-----------

Figure 7 HEZB TABLE

3.5.7 Word And Phrase Search

This routine prepares the information needed to display the text required by the user (the Aiah that contains the word or phrase). The user specifies if the search is to be performed in a specific Sora, Juz, or the whole text of the Holy Quran. Then, the display text routine handles the rest of the operation.

A table (word table) is established to enhance the operation of the routine. This table contains the most frequently used words in the Holy Quran. Each entry of the table is a record which contains the following data:

- a. the word text (used to identify the word),
- b. Juz number (used to specify the file name and for information display),
- c. line number in Juz (file) (used to specify the exact location of the word in the file for direct access),
- d. Sora number (used for information display),
- e. Aiah number (used for information display), and
- f. pointer to the next position (used to specify the next position of the word in the Holy Quran).

The next position contains the following data:

- a. Juz number,
- b. line number in Juz (file),

- c. Sora number,
- d. Aiah number, and
- e. pointer to the next position.

The most frequently used words are stored in the table. Any data needed are obtained from the Sora and Hezb tables. The routine performs sequential search if the search is to be performed in a Sora or a Juz. If the whole Quran search option is chosen, the routine uses the word table to perform the operation.

Figure 8 shows the structure of the table.

The module consists of the following routines:

1. the main procedure,
2. sequential search, and
3. direct search routines.

The main procedure obtains the user's option and calls the direct-search routine if the option is to search in the whole Quran. Otherwise, it will call the sequential-search routine. If the direct-search routine does not find the required text in the word table, the sequential-search routine will be called.

The block diagram of this routine is shown in Figure 9.

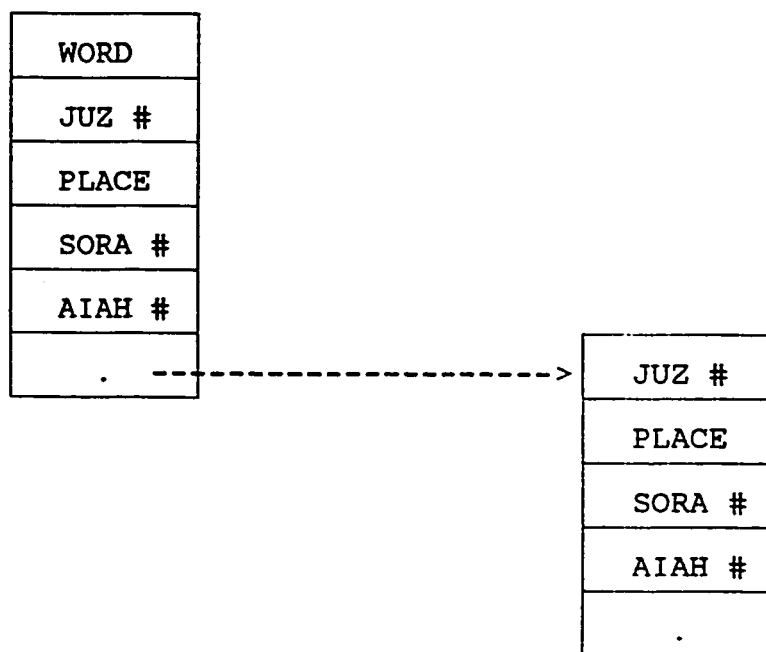


Figure 8 WORD TABLE

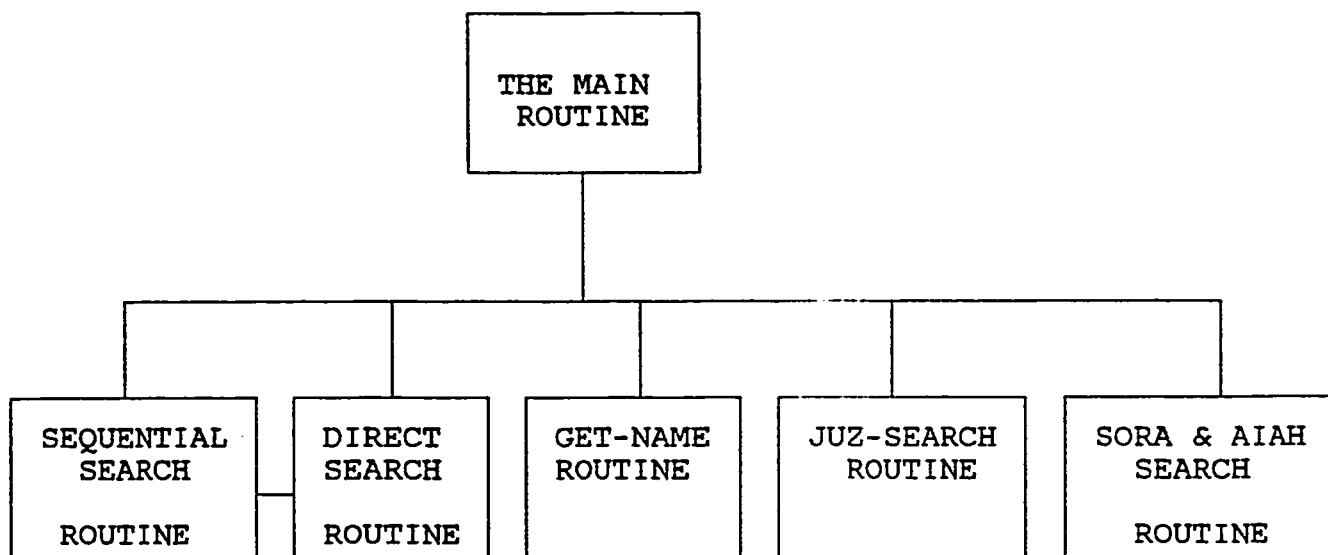


Figure 9 BLOCK DIAGRAM OF THE WORD-AND-PHRASE-SEARCH ROUTINE

3.5.7.1 The Main Procedure

The procedure displays a menu which has four options: search in a Sora, search in a Juz, search in the whole Quran, and exit the routine. Then, calls to other routines continue the rest of the operations. The procedure has a parameter indicating a word or phrase search.

The procedure performs the following operations :

- loop1
- display the options of the menu (search in a sora, Juz, Quran, or exit),
- if the option is to search in a Sora, then :
 - a call Sora-and-Aiah-search routine with a parameter indicating a Sora search (to open the files and position the file pointers at the beginning-of-Sora),
- else if the option is to search in a Juz, then :
 - a call to the Juz search routine (to open the files and position the file pointers at the beginning-of-juz),
- if it is a word search (passed as a parameter), then :
 - begin1
 - display a message asking the user to enter a

```

word,
- read the word by calling get-name routine,
- if the user has entered a phrase (more than one
  word), display an error message and exit to the
  search menu,
- end1,
- else if it is a phrase search, then :
  - begin2
  - display a message asking the user to enter a
    phrase,
  - read the phrase by calling get-name routine,
  - end2,
- if the option is for a search in the whole Quran,
  call the direct-search routine,
- else, call the sequential-search routine,
- if the option is to exit the routine, quit the
  loop,
- endloop1.

```

3.5.7.2 Sequential Search Routine

This routine searches for a specific word or phrase sequentially in a specific Sora, Juz, or the Quran. The Aiah in which the first occurrence of the required word or phrase is displayed to the user. Then, if the

PgDn key is pressed, the rest of the Aiah's are displayed. If the CR key is pressed, the display is terminated. If the space bar key is pressed, the next occurrence of the word or phrase is displayed if it occurs. The routine has two parameters, the first is holding the word or the phrase and the second indicates the option entered by the user in the main procedure.

The routine performs the following operations :

- loop1
- loop2
- read the next line from the characters and accentuation marks files, (the files are already opened and positioned to the required place),
- if the required word or phrase is found in the line, quit loop2,
- else
 - begin1
 - if the line does not contain a full Aiah, read the rest of the Aiah and check if the word or phrase is found while reading,
 - if the user's option is to search in a Sora, then :
 - begin2
 - if the end-of-sora or the end of the Quran

```
are reached, quit loop2,
- if the end-of-file is reached, then:
  - begin3
  - close the current opened files,
  - if the end of the Quran is reached, quit
    loop2,
  - else,
    - begin4
    - increment the Juz number,
    - check if the next files are found. If
      not found, then exit the routine,
    - else, open the files,
    - end4,
  - end3,
- if the end-of-hezb is reached then increment
  Hezb number,
- end2,
- if the user's option is to search in a Juz,
  then :
  - begin5
  - if the end-of-file is reached, quit loop2,
  - if the end-of-sora is reached, increment
    Sora number,
  - if the end-of-hezb is reached then increment
```

```
        Hezb number,
    - end5,
- if the user's option is to search in the whole
  Quran, then :
    - begin6
    - if the end-of-file is reached, then
      - begin7
      - close the current opened files,
      - if the end of the Quran is reached, quit
        loop2,
      - else
        - begin8
        - increment the Juz number,
        - check if the next files are found. If
          not found, then exit the routine,
        - else, open the files,
        - end8,
      - end7,
    - if the end-of-sora is reached, increment
      Sora number,
    - if the end-of-hezb is reached then increment
      Hezb number,
    - end6,
- endloop2
```


- if the word or phrase is not found, display a message indicating that no more occurrences of the word or phrase and exit the routine,
- call the graphics and contextual analysis routine to display the Aiah that contains the required text,
- display Aiah number,
- prompt for a user's input,
- if the key pressed is PgDn, call the contextual analysis routine to continue the display operation,
- else if the key pressed is a CR, quit loop1,
- else if the key pressed is a space bar, repeat loop1 again,
- endloop1.

3.5.7.3 Direct Search Routine

This routine searches for a specific word or phrase, using the word table, in a specific Sora, Juz, or the Quran. The Aiah in which the first occurrence of the required word or phrase exists is displayed to the user. Then, if the PgDn key is pressed, the rest of the Aiah's are displayed. If the CR key is pressed, the display is terminated. If the space bar key is

pressed, the next occurrence of the word or phrase is displayed if it occurs. The routine has two parameters, the first is holding the word or the phrase and the second indicates if the search is for a phrase or a word. The routine is called if the user requires a search in the whole Holy Quran.

The routine performs the following operations :

- loop1
- if the option is to search for a phrase, then:
 - begin1
 - get the next word in the phrase,
 - if the the word is the last one in the phrase,
set a flag,
 - search for the word obtained in the word table
and set a flag if found,
 - end1,
- else search for the word passed as a parameter in
the word table and set a flag if found,
- if the word is not found in the table and the
option is to search for a word, then:
 - begin2
 - display a message asking the user if the
spelling of the word or phrase entered is
correct,

- .
- if the answer is no then exit the routine to let the user to try the operation for another time,
- otherwise, open the first files in the Holy Quran (Juz 1) if they exist,
- if they do not exist, display an error message and exit the routine,
- call the sequential-search routine and exit the routine upon return,
- end2,
- else
 - begin3
 - obtain the information about the word from the word table,
 - loop2
 - open the files where the word or phrase occurs,
 - position the file pointer at the start of the line in which the word or phrase occurs, (repeat READLN number of times found in the table as a position of the word),
 - if it is a phrase search, then :
 - begin4.
 - read the line,
 - if the phrase is found in the line, set a

```
        flag,
    - end4,
- else close the files and exit loop3,
- if the word or phrase is found in the line,
then :
    - begin5
    - read the rest of the Aiah if it occurs in
      more than one line,
    - call the graphics and contextual analysis
      module to display the Aiah,
    - display the Aiah number,
    - prompt for a user's input,
    - if the key pressed is PgDn, call the
      contextual analysis routine to continue the
      display operation,
    - else if the key pressed is a CR, quit loop1
      and loop2,
    - else if the key pressed is a space bar,
      then:
        - begin6
        - close the currently opened files,
        - if there is another occurrence for the
          word, update the information about the
          word and repeat loop2 again,
```

```
- else quit loop2,  
- end6,  
- end5,  
- endloop2,  
- end3,  
- endloop1.
```

3.5.8 Display Text

The function of this routine is to display the text of the Holy Quran starting from a specific position. It is established for the search routines discussed above. The routine interacts directly with the graphics and contextual analysis module. It consists of the following subroutines :

- a. display frame,
- b. display information inside the frame,
- c. display a line,
- d. display until end-of-Juz, and
- e. display until end-of-Quran.

The display until-end-of-Quran is the main routine that calls the other routines. It makes sure that the display operation continues until the end of the Holy Quran or stopped by the carriage return (CR) key. It calls the display until end-of-juz routine that displays text until the end of the Juz. This routine calls display-a-line

routine which makes sure that the line is displayed on the screen. The display frame and display information inside the frame are called at the beginning and called each time a new screen is to be displayed. In addition, if new information occur, the display information inside the frame is called. The carriage return (CR) key terminates the display operation and the (PgDn) key continues the display. The block diagram of the routine is shown in Figure 10.

3.5.8.1 Frame Display

This routine is simple. It displays the frame of the screen that fits the requirements to display full information about the text displayed. The routine divides the screen into two parts, one for text and the other for the information by drawing straight lines.

3.5.8.2 Display Information Inside The Frame

This routine displays full information about the Holy Quran text displayed on the screen. It displays the information inside the area specified for that on the screen. The information is displayed in one line only.

The routine displays the following information :

- a. Sora number,

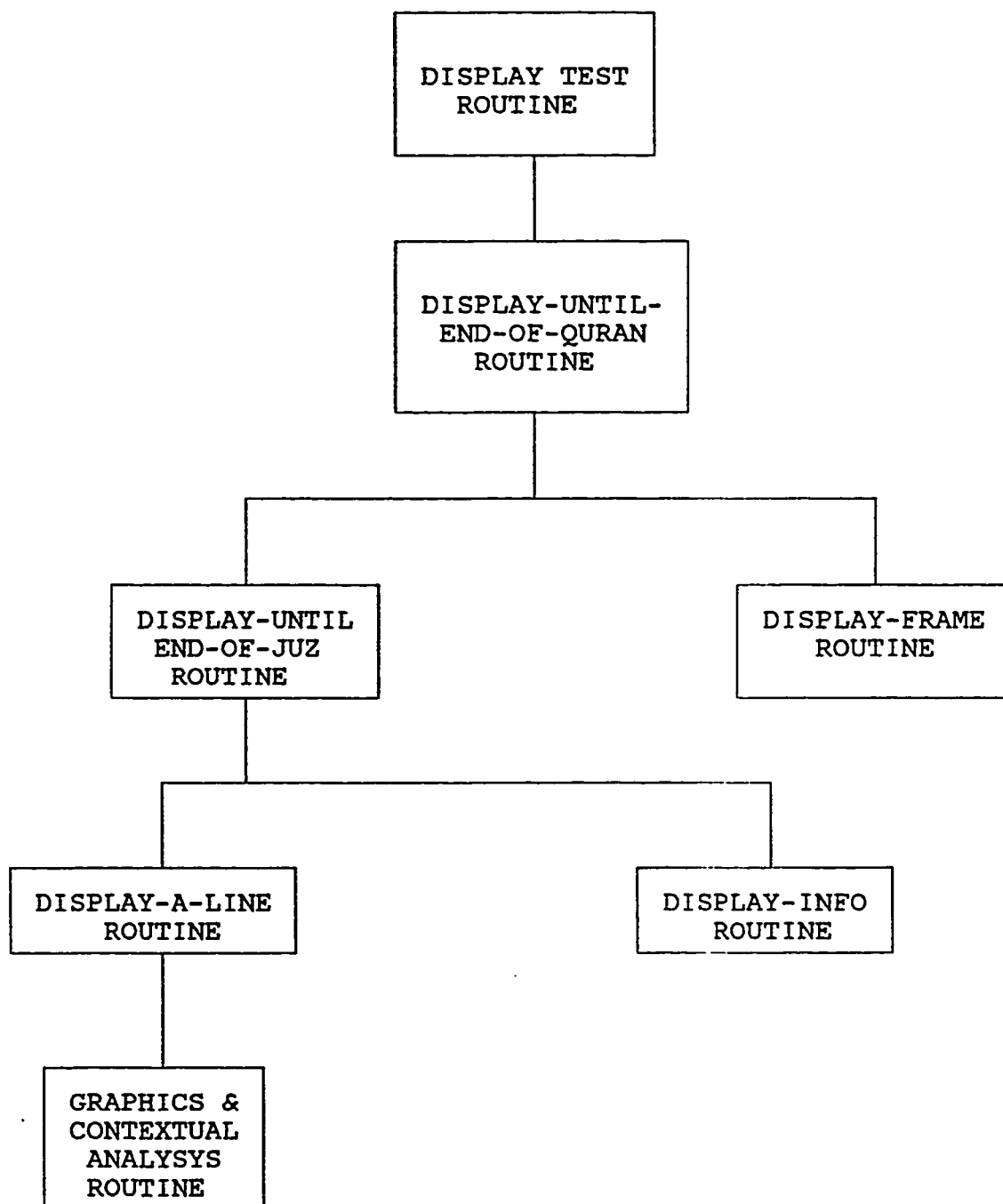


Figure 10 BLOCK DIAGRAM OF THE DISPLAY TEST ROUTINE

- b. Sora name,
- c. number of Aiah's in Sora,
- d. Sora classification,
- e. Juz number, and
- f. Hezb number.

It obtains the information from the tables discussed in the search routines. The following operations are performed by the routine :

- display the Sora number at the beginning of information line by calling the draw number routine of the graphics and contextual analysis module,
- display the Sora name obtained from the Sora table, and call the contextual analysis routine,
- display the number of Aiah's in Sora obtained from the Sora table and call the draw number routine,
- display the Sora classification obtained from the Sora table and calls the contextual analysis routine,
- display the Juz number augmented with the 'جزء' word by calling the contextual analysis routine,
- display the Hezb number augmented with the 'حزب' word by calling the contextual analysis routine,
- if it is the beginning of a Sora then display the Basmalah text on the next line and if the Sora

number is 1 then display Aiah number 1 after the Basmalah text.

- advance the line number to the next line for later display.

3.5.8.3 Display A Line

This routine displays all the text found in the variables passed as parameters (text variable and accentuation mark variable). The display is of a good Al-Rasm Al-Othmani style of the Holy Quran text. The variables passed as parameters may contain one or part of an Aiah.

The following operations are performed by the routine :

- obtain the next word in text,
- obtain the corresponding accentuation mark,
- if the word (approximately) does not fit in the current line, then
 - begin1
 - advance to the next line and set the column to the beginning of the line,
 - if the next line does not fit in the current screen, then
 - begin2
 - wait for a user input,

- if the key pressed is the (PgDn) key then
erase the screen, call the display frame,
and display information inside the frame
routine,
- if the key pressed is the carriage return
(CR) key then exit the routine,
- end2,
- endl,
- call the contextual analysis routine to display
the text and the accentuation marks.
- if the blank following the word has an
accentuation mark assigned to it, then draw this
mark,
- if not then advance one space,
- continue the operation until the end-of-aiah
character is found or all the text is displayed.

3.5.8.4 Display Until End-Of-Juz

This routine displays all the text found in the files (Juz) opened and specified by the calling routine. The routine reads a specific number of lines from the files in a buffer and calls the display-a-line routine for each line in the buffer. The routine checks if a new Sora or a new Hezb are reached and reflects it

in the information line displayed on the screen.

The following operations are performed by the routine :

- call the display information routine,
- read the text and accentuation marks from the file already opened and store them in two separate buffers (one for text and the other for accentuation marks),
- if the end-of-file is reached while reading, then stop reading and set a flag accordingly,
- repeat the following operations for the number of lines read,
- if it is the end of the current Sora, then
 - begin1
 - wait for a user input (continue or exit),
 - erase the screen and display the frame and its information,
 - end1,
- if it is the end of the current Hezb, then
 - begin2
 - increment the Hezb number and display the new value on the screen (specified position in the information line),
 - end2,
- call the display-a-line routine to display a line

- of text in the buffer,
- if an end-of-aiah character is found in the line, then
 - begin3
 - if there is a place in the current line of display on the screen enough for the display of the Aiah number, then display it,
 - otherwise, advance to the next line and set the column at the beginning of the line,
 - if the screen is full, then wait for a user input and either continue the display on the next screen or terminate the operation,
 - end3,
 - increment the Aiah number if the end-of-aiah character found in the line,
 - increment the index of the current line in the file, and
 - end of repetition.

3.5.8.5 Display Until End-Of-Quran

This routine assures the continuity of the display process until the end of the Holy Quran or user's request to stop the operation. It closes the previously opened files and open the following ones in

the same order found in the Holy Quran. The display process continues until the end of Quran or user's termination of the process. It is considered as the main routine in the display text routine.

The following operations are performed by the routine :

- loop1
- call the display frame routine,
- call the display until end-of-Juz routine, (it displays the text until the end of the current opened file),
- wait for a user response, either to continue displaying or to terminate,
- if the (PgDn) key is pressed, then
 - begin1
 - if the Juz number is less than the maximum (30), then
 - begin2
 - close the current files,
 - increment the Juz number,
 - open the new files (augment the Juz number with text to obtain the file names),
 - if the files do not exist, then display an error message and exit the routine,
 - otherwise, open the new files and set the

```

        index on the file position to equal 1,
    - end2,
- else (the Juz number greater than 30)
    - begin3
    - display a message indicating that the end of
      the Holy Quran has been reached,
    - exit the routine,
    - end3,
- end1,
- if the key pressed is a (CR) then exit the
  routine,
- endloop1.

```

3.6 Tajweed Rules Module

Tajweed rules are a set of rules used to aid the reader of the Holy Quran to pronounce in the proper way.

The function of this module is to analyze a specific Aiah to obtain these rules (explained below). The module interacts with the search module to use the search for an Aiah part. In addition, it interacts with the graphics and contextual analysis module to display the Aiah, highlight the letter which is applicable for a rule, and display messages and information. The module is activated from the user interface module upon user request. It interacts

indirectly with the storage module via the search module.

Tajweed rules generated by this module consist of subsets of rules [6]. They are :

- Meem Sakenah rules,
- Noon Sakenah and Tanween rules,
- Lam rules,
- Rah rules,
- Meem and Noon Mushaddadain rules,
- Madd rules,
- Mutamathelain rules,
- Mutajanesain rules,
- Mutaqarebain rules,
- Ha' rules,
- beginning-of-Sora rules,
- Waqf rules,
- letters rules, and
- special cases rules.

Each of these rules is checked and the corresponding message is displayed to the user on the screen.

This module provides two options to the user :

- a. display rules for a specific Aiah. It performs the above mentioned rules and generates the messages to the user.
- b. display information about the rules. It displays the

meanings of the rules displayed if the user needs information about them.

The module has three parts (routines), which are :

1. the main routine,
2. display information about Tajweed rules, and
3. analyze the Aiah.

The main routine branches to the routine specified by the user's option (either to analyze the Aiah for rules or to display information about them). To analyze an Aiah, the Aiah is retrieved and displayed by calling the search and the graphics and contextual analysis modules.

The block diagram is shown in Figure 11.

3.6.1 The Main Procedure

The routine displays a menu which has three options :

1. display information about Tajweed rules,
2. analyze an Aiah for Tajweed rules, and
3. exit the module.

Upon user's request, the corresponding routine is invoked unless the request is to exit the module. The main procedure invokes the graphics and contextual analysis module to display the menu and to get the user's request.

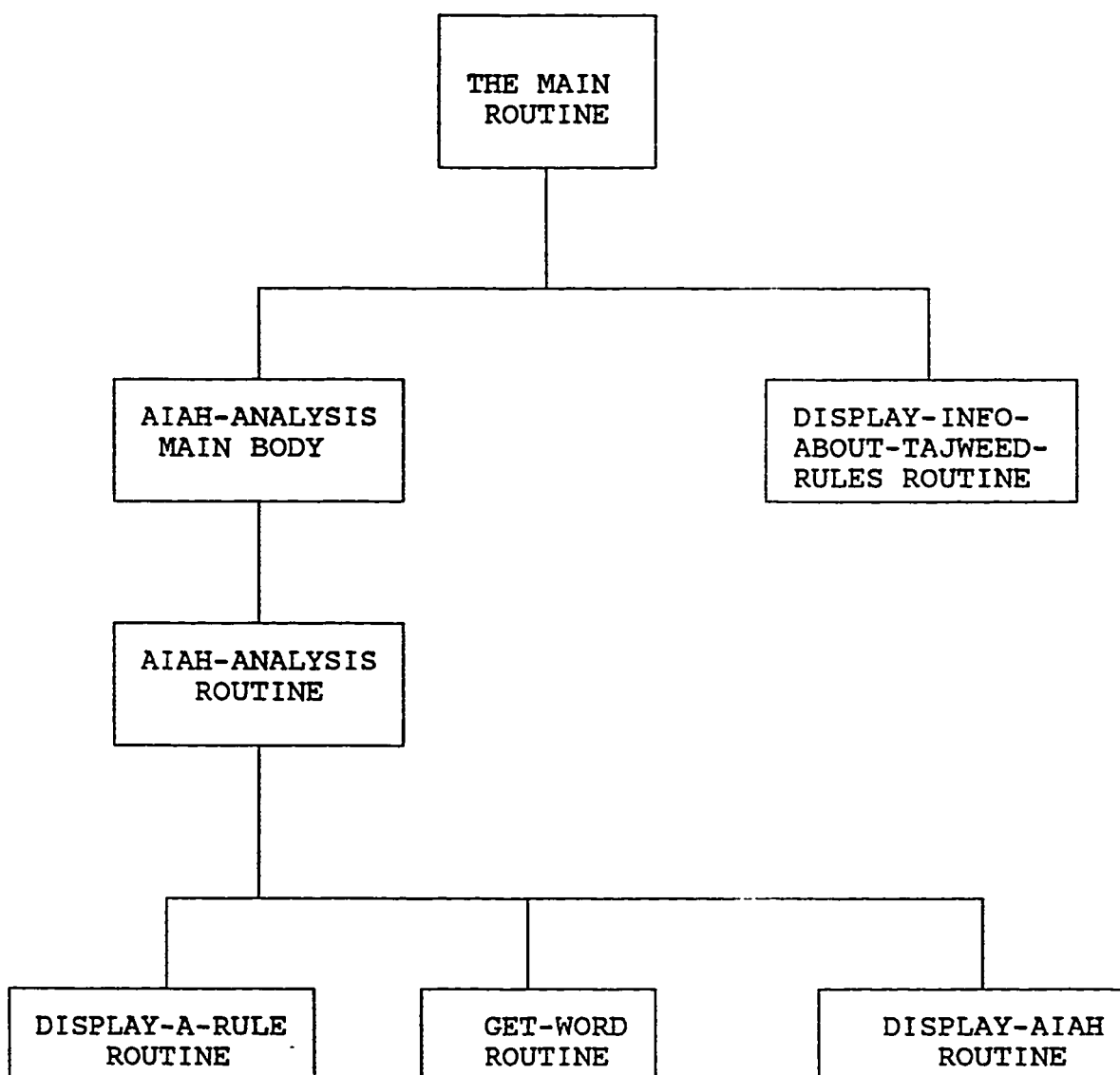


Figure 11 BLOCK DIAGRAM OF TAJWEED RULES MODULE

3.6.2 Display Information About Tajweed Rules

This routine displays the basic definitions of the Tajweed rules. This is useful for users that need to learn Tajweed or who do not know the basics of Tajweed rules.

It is a simple routine that displays messages of information (definitions) on the screen. The routine displays a menu of choices. The user is able to choose any rule to get information about it by entering a number specifying the rule. The routine returns to the menu after displaying the information needed by the user until exit is chosen. Each time the screen is full of information, the routine prompts for an input from the user. Either to quit or to continue using the CR and PgDn keys, respectively.

3.6.3 Aiah Analysis

The function of this routine is to analyze the Aiah specified by the user for the Tajweed rules and to display these rules to the user. For each rule found the corresponding character is highlighted. To perform this operation, the graphics and contextual analysis is used. The routine has the facility to continue the analysis for the following Aiah upon user's choice.

This routine consists of the following routines :

- a. the main procedure,
- b. display Aiah,
- c. get a word from the Aiah,
- d. display a rule, and
- e. rules analysis.

The main procedure prompts for a user's input for specified choices. It is able to continue the analysis for the next Aiah as needed by the user. The routine calls the rules-analysis routine to analyze the Aiah for Tajweed rules. The rules-analysis routine calls the search module to retrieve the specified Aiah. After that, it calls the display-Aiah routine to display the chosen Aiah. It uses the get-word and display-a-rule routines during the analysis process.

3.6.3.1 The Main Procedure

The procedure retrieves the user's options that give the user the choice to display detailed Tajweed rules or the basics, and the choice of pressing a key or waiting for a specific time after each display of a rule. It calls the rules-analysis routine to display the rules. A choice is given to the user to continue the process for the next Aiah or to exit the module.

The routine performs the following operations :

- call the Sora-and-Aiah- search routine in the search module,
- retrieve the first option (mentioned above) and set a flag accordingly,
- retrieve the second option (mentioned above), and set a flag accordingly,
- call the rules-analysis routine,
- prompt the user to continue the analysis for the next Aiah or to exit the routine, (CR means exit and PgDn means continue).

3.6.3.2 Display Aiah

This routine displays the text of the Aiah needed for analysis. The display does not continue for the following Aiah's as in the search module. It stops at the end of the Aiah. In addition, the routine displays the rules applicable.

The routine performs the following operations:

- display a frame and the information inside the frame using the routines of the search module,
- loop1
 - obtain the next word of the Aiah,
 - obtain the corresponding accentuation marks,
 - display the word using the contextual analysis

- module,
- save the shapes of the characters while displaying (for later use),
 - save the position of the end-of-line if reached (used later for highlighting the characters)
 - repeat the above operations for the rest of the text in the Aiah,
 - if end-of-aiah character is found, then exit the loop and display the aiah number.
 - endloop1.

3.6.3.3 Get A Word

The routine obtains the next word of the Aiah text including the corresponding accentuation marks. The routine is useful for the analysis of special case rules and for the beginning of Sora cases.

The following operations are performed :

- store in a character-by-character fashion the following characters in the Aiah. This is done by repeating the operation until a blank or an end-of-aiah character is found, and
- store the corresponding accentuation marks.

3.6.3.4 Display A Rule

The routine displays a specific rule passed as a parameter. It simply displays the rule in a specific part on the screen. Then, it either waits for a key press or waits for a specific amount of time. It depends on the user's choice. After that, the routine erases the message displayed.

3.6.3.5 Rules Analysis

The function of this routine is to analyze the Aiah specified by the user for the Tajweed rules and to display these rules to the user. For each rule found, the corresponding character is highlighted. To perform this operation, the graphics and contextual analysis is used.

The Aiah analysis routine performs the following operations :

- initializations,
- if the Aiah is a beginning of a Sora, then :
 - call get-a-word to analyze the beginning of Sora special cases, and
 - set a flag.
- check for the other special cases and set a flag as an indication,

- move the cursor to the following character (used for highlighting characters later),
- if the end of the line is reached, then move to the next line,
- if the end of the screen is reached (last character in the screen), then call the display text routine to display the next screen,
- obtain the previous, and next few characters using the array of text,
- At the same time, obtain the corresponding accentuation marks,
- if the end-of-aiah character is reached while storing the characters, then store blanks instead,
- if the next character is a blank, then obtain the one after and set a flag indicating that it is the end of a word,
- now the Tajweed symbols are checked by checking if the text character is a blank and the corresponding accentuation mark is not,
- according to the accentuation mark found, the rule is displayed by calling the display routine (the rules concerning with stoppings),
- if the character is not a blank, then :
 - if the user chooses to have the detailed

information about Tajweed rules, then (rules about the letter itself)

- highlight the letter on the screen using the shapes stored earlier,
- the letter is checked if it is a member of specific groups defined for Tajweed and determines the position from where the letter is pronounced inside the mouth,
- a message is displayed for each occurrence of the letter in a group, and
- at the end the routine either waits for a key press or waits for a specific amount of time (depending on the user's choice),
- the special begin-of-sora case is handled,
- if the word obtained is a member of the special words table, then :
 - check the characters of the word and display the rule that applies, and
 - set the flag back if the end-of-word is reached or the word is not found on the table,
- now, the current letter is considered for the rest of the rules,
- the current letter is checked with the adjacent

letters,

- if any rule is applied, then the rest of the rules are ignored, since one rule can be applicable at one time,
- the corresponding letter is highlighted, a sound is activated, and the rule is displayed, and
- the operations are repeated again for the rest of the letters of the Aiah.

The system is implemented using TURBO PASCAL V4.0. The reasons for choosing it are :

- a. it has good graphics capabilities,
- b. it has the data structure capabilities,
- c. it is efficient for the tables needed in the system,
and
- d. it is suitable for large systems development since
it allows the executable code to be larger than 64K
bytes through the use of UNITS.

The system consists of three programs. They are :

1. the shape generation,
2. the data entry, and
3. the main program that consists of all of the rest of
the modules.

The sizes of these programs are 18967, 64896, and , 123424 bytes respectively.

A user manual written in Arabic is shown in Appendix A.

Furthermore, a user manual for the data entry personnel written in Arabic is shown in Appendix B.

Due to the length of the program listings, it is decided not to include them in the thesis. If needed, please contact the author.

Chapter IV

CONCLUSION AND FURTHER WORK

This research has presented the design and implementation of a retrieval and display system for the Holy Quran. This research has modified and added some of the shortcomings of the other systems. In addition, it has its own advantages.

The design of the system depends on organizing it into parts which interact with each other. This means it is modular and provides the ability to add or delete any part with minimum effect on the other parts.

These parts are summarized below.

1. The user interface module which is the interface between the user and the system. The module accepts the user's request, and invokes the corresponding module that generates the output to the user.
2. The graphics and contextual analysis module is used to display menus, information, messages, and the Holy Quran text all in Arabic. The letters, numbers, and other shape are displayed using graphics. A contextual analysis routine is used to display the right shapes. The shapes are generated using a separate routine

developed for this process.

3. The storage module specifies the organization of the Holy Quran's data stored. The letters are stored in a file and the accentuation marks are stored in another one. Text files are used for data storage. The data are stored in lines. The line may consist of an Aiah or a part of an Aiah depending upon the Aiah size.
4. The data entry module is used to aid in the data entry process. A simple type of editor is implemented to make the process easier for the data entry personnel.
5. The search module obtains the data needed by the user. A multi-index search is performed by the module. It searches for a specific:
 1. word,
 2. phrase,
 3. Aiah,
 4. Sora,
 5. Juz, and
 6. Hezb.

Then, the text is displayed. The user can continue the display until the end of the Holy Quran or user termination of the process.

6. Tajweed rules module displays the rules applicable on an Aiah specified by the user. In addition, it

displays the definitions of these rules if the user requests that. The routine generates the basic rules that every reader of the Holy Quran needs to know.

The research provides the basis for further additions and modifications. The system can be further developed by adding new functions. These functions are:

- a. the meanings of the words in the Holy Quran,
- b. the Holy Quran explanatory notes (Tafseer),
- c. reasons for Nozool of some Aiah's,
- d. the way to pronounce some words (written differently), and
- e. vocal recitation of the Holy Quran with the different "readings".

We hope that this research would be a corner stone for further research and development on the subject.

REFERENCES

- [1] Salsabeel, Applied Microsystems Technology Limited, 32-34 Circklewood Broadway, London, NW2 3ET, United Kingdom.
- [2] The Holy Quran, National Group Company, P.O.Box 2349, Riyadh, Saudi Arabia.
- [3] The Holy Quran, Al-Alamiah, P.O.Box 5954, Riyadh 11432, Saudi Arabia.
- [4] M.G.Khayat., "Printing Arabic Text Using Dot Matrix Printers", Software-Practice and Experience, Vol. 16 (2) 165-172 (February 1986).
- [5] Sayed S. Hyder and F. Richer, "The Theory And Design OF A System For Printing And Communication in Arabic-Farsi-Urdu Languages, Biosci, Communication, 3, 181-206 (1977).
- [6] Izzat Daas, Fann Al-Tajweed, Maktabet Al-Ihsan, Ninth Edition (1982).

Appendix A
USER MANUAL

بسم الله الرحمن الرحيم

نظام لعرض وتخزين واسترجاع القرآن الكريم

دليل المستخدم

بسم الله الرحمن الرحيم

نظام لعرض وتخزين واسترجاع القرآن الكريم

المقدمة :

يعتبر هذا الدليل بمثابة المرشد للأشخاص الذين يرغبون بإستخدام هذا النظام . ويمكن استخدام هذا الدليل من قبل أي شخص بغض النظر عن مدى معرفته بالحاسبات الآلية . واستخدام النظام سهل للغاية بحيث يمكن لبعض المستخدمين استعماله من غير قراءة هذا الدليل .

ويتميز النظام بما يلي :

- ١ - عرض نصوص القرآن الكريم بالرسم العثماني .
- ٢ - امكانية البحث عن كلمة ، أو مقطع من آية ، أو حزب ، أو جزء ، أو آية في سورة أو سورة .
- ٣ - عرض الاحكام المتعلقة بالتجويد في آية آية في القرآن الكريم .
- ٤ - امكانية طبع النص القرآني على الطباعة .
- ٥ - استخدام النظام باللغة العربية .
- ٦ - سهولة استخدام النظام .
- ٧ - استخدام النظام على أجهزة أي بي إم الشخصية أو المطابقة لها .

ويتألف الدليل من أجزاء مختلفة يمكن قراءة أي قسم على حدة ما عدا القسم الخاص بكيفية البدء بتشغيل النظام الذي يجب أن يقرأ في البداية .

بدء تشغيل النظام :

إذا كان الجهاز الذي تقتنيه يحتوي على إسطوانة صلبة (Hard disk) فمن المفضل أن تنقل المعلومات الموجودة في الإسطوانات الممغنطة (Floppy disks) إليه . راجع الدليل الخاص بالـ (DOS) لمعرفة ذلك . وإذا أردت أن تستخدم الطابعة لطباعة نص القرآن الكريم يجب عليك أن تدخل الأمر التالي الخاص بالـ (DOS) :

A>GRAPHICS

والآن يمكنك تشغيل النظام بإدخال :

A>QURAN

وستظهر لك الشاشة الموجودة في الشكل ١ . وهي عبارة عن اللوحة الرئيسية للنظام .

اللوحة الرئيسية :

تحتوي اللوحة الرئيسية على أربعة إختيارات كما هو مبين في الشكل ١ . ويتقبل النظام إحدى هذه الأرقام (١-٤) فقط . ومن ثم ينتقل النظام إلى شاشة جديدة بحسب الرقم المدخل . ويمكن الخروج من النظام بإدخال الرقم ٤ . باستخدام الرقم ١ يمكن عرض الفهرس للقرآن الكريم . ورقم ٢ ، تمكن المستخدم من البحث في القرآن الكريم باستخدام أصوات مختلفة . وأخيراً الرقم ٣ ، يعطي المستخدم إمكانية في استخراج أحكام التجويد للآيات .

١ - الفهرس :

إذا أدخل المستخدم الرقم ١ في اللوحة الرئيسية تظهر له شاشة تحتوي على الفهرس للقرآن الكريم . ويبين الشكل ٢ هذه الشاشة . وتحتوي الشاشة على رقم السورة ، واسم السورة ، ورقم

اللوحة الرئيسية

١ الفهرس

٢ البحث

٣ احكام التجويد

٤ الخروج

الاختيار :-

رقم	اسم السورة	الجزء	الحزب	الآيات
١	الفاتحة	١	١	٧
٢	البقرة	١ إلى ٣	١ إلى ٥	٢٨١
٣	آل عمران	٣ إلى ٤	٥ إلى ٨	٢٠٠
٤	النساء	٤ إلى ٦	٨ إلى ١١	١٧٦
٥	المائدة	٦ إلى ٧	١١ إلى ١٣	١٢٠
٦	الأنعام	٧ إلى ٨	١٣ إلى ١٥	١٦٥
٧	الأعراف	٨ إلى ٩	١٦ إلى ١٨	٢٠٦
٨	الأنفال	٩ إلى ١٠	١٨ إلى ١٩	٧٥
٩	التوبة	١٠ إلى ١١	١٩ إلى ٢١	١٢٩

الجزء ، ورقم الحزب ، وعدد الآيات في السورة . ويكون العرض لشاشة واحدة فقط ومن ثم يستطيع المستخدم الإستمرار أو الخروج . فإذا أراد الإستمرار ، يضغط على المفتاح (PgDn) على لوحة المفاتيح . أما الخروج فيكون بالضغط على المفتاح (CR) أو (لـ) وبعد الخروج يرجع النظام إلى اللوحة الرئيسية مرة أخرى . ويمكن الإستمرار في العرض حتى نهاية الفهرس ومن ثم يرجع النظام تلقائيا إلى اللوحة الرئيسية .

٢ - البحث :

عند اختيار الرقم ٢ في اللوحة الرئيسية تظهر لوحة أخرى (شاشة) تحتوي على عدة اختيارات توضح الأساس في عملية البحث بالإضافة إلى اختيار الخروج من البحث . وشكل ٢ يبين شاشة البحث هذه . ويتقبل النظام الاعداد (١ - ٧) لإكمال العمليات الخاصة بالبحث وإذا أدخل الرقم ٧ يتم الخروج إلى اللوحة الرئيسية .

١-٢ البحث عن كلمة أو مقطع من آية :

عند اختيار الرقمين ١ أو ٢ (البحث عن كلمة أو مقطع من آية) تظهر لوحة جديدة تحتوي على أربعة اختيارات تحدد عملية البحث كما هو موضح بالشكل ٤ . ويتقبل النظام الاعداد من (١ - ٤) والرقم ٤ عبارة عن الخروج إلى اللوحة السابقة .

١-١-٢ في سورة معينة :

إذا حدد المستخدم الرقم ١ (أي البحث في سورة معينة)، يقوم النظام بالطلب من المستخدم اسم السورة والتي يتم إدخالها من قبل المستخدم . إذا لم توجد اسم السورة في النظام ، تظهر عبارة أسفل الشاشة توضح ذلك وتطلب اسم السورة مرة ثانية . في حين وجود السورة ، يطلب من المستخدم إدخال

البحث عن

١ كلمة

٢ مقطع من آية

٣ حزب

٤ جزء

٥ آية

٦ سورة

٧ الخروج

الاختيار من

١ في سورة معينة

٢ في جزء معين

٣ في القرآن الكريم

٤ الخروج

الاختيار :-

الكلمة أو المقطع بحسب الاختيار في لوحة البحث .
 انظر الشكل ٥ . في حالة إدخال كلمة ، فإن
 النظام لا يقبل فراغا في الكلمة أو كلمة أكبر من
 ٣٠ حرفا . أما بالنسبة للمقطع فإن النظام لا يقبل
 أكثر من (٤٠) حرفا . بعد ذلك يقوم النظام بالبحث
 عن الكلمة أو المقطع في السورة المعينة ويعرض
 الآية الموجودة بها النص المطلوب (شكل ٦) . في
 حالة عدم وجود النص يقوم النظام بالعودة إلى
 اللوحة السابقة (شكل ٤) . وإذا أراد المستخدم
 البحث عن الكلمة نفسها في موضع آخر في السورة
 يضغط على مفتاح المسافة في لوحة المفاتيح ويتم
 عرض الآية الموجودة بها النص المطلوب إن وجد أو
 الرجوع إلى اللوحة السابقة . في حالة الرغبة في
 الإستمرار بعرض آيات القرآن الكريم يتم ذلك بالضغط
 على مفتاح (PgDn) بعد عرض كل شاشة . وفي حالة
 الرغبة بالتوقف عن العرض أو الإستمرار في البحث
 يضغط على مفتاح (CR) .

٣-١-٢

في جزء معين :

عند اختيار الرقم ٢ (أي البحث في جزء معين) ،
 تظهر شاشة جديدة ويطلب من المستخدم إدخال رقم
 الجزء (بين ١ - ٣٠) . ثم يطلب النص المطلوب
 البحث عنه وتستمر العملية كما وضح في الفقرة
 السابقة تماما . (شكل ٧) .

٣-١-٢

في القرآن الكريم :

يتم البحث في القرآن الكريم كله إذا اختار
 المستخدم الرقم ٣ . وتظهر شاشة جديدة تطلب من
 المستخدم النص المطلوب البحث عنه . ومن ثم تستمر
 العملية كما ذكر في فقرة (٢-١-١) . (شكل ٨) .

أدخل اسم السورة : الدبا

أدخل الكلمة : الفجلـ

(un) النبا (ع = ع) مكية جزء = ٣ حزب ٥٩

إِنَّ يَوْمَ الْفَصْلِ كَانَ مِيقَاتُ (١٧)

ادخل رقم الجزء : ٣

ادخل الكلمة : الدين

ادخل الكلمة: الخاضعات

(شكل - ٨)

٢-٢ البحث عن جزء أو حزب :

إذا تم اختيار الرقمين ٢ أو ٤ (بحث عن جزء أو حزب) من لوحة البحث يقوم النظام بعرض شاشة جديدة تطلب من المستخدم رقم الحزب (١-٦٠) أو رقم الجزء (١-٣٠) (شكل ٩ ، ١٠) . إذا ادخل الرقم خارج هذا النطاق تظهر عبارة تحذيرية للمستخدم ويطلب منه الإدخال مرة أخرى . بعد ذلك ، يقوم النظام بالبحث المطلوب وعرض النص من بداية الاسم في البحث لشاشة كاملة من النص . وإذا أريد الإستمرار في عملية العرض يضغط على مفتاح (PgDn) وعلى مفتاح (CR) لوقف العملية والرجوع إلى لوحة البحث ويمكن الإستمرار في عملية العرض حتى نهاية المصحف أو الخروج بالضغط على مفتاح (CR) .

٢-٢ البحث عن آية أو سورة :

عند اختيار الرقمين ٥ أو ٦ (بحث عن آية أو سورة) من لوحة البحث ، يقوم النظام بعرض شاشة جديدة تطلب من المستخدم إدخال اسم السورة . في حالة إدخال الاسم خطأ تعرض عبارة تحذيرية أسفل الشاشة ويطلب اسم السورة مرة أخرى . إذا كان البحث عن آية ، يطلب رقم الآية أيضا (شكل ١١ ، ١٢) . وإذا وجد أن الرقم المدخل خارج نطاق السورة (أكبر من عدد آيات السورة) يتم عرض عبارة تحذيرية ويطلب الرقم مرة أخرى . ثم يقوم النظام بعرض النص من بداية الاسم في البحث. راجع الفقرة (٢ - ٢) للمزيد من المعلومات عن العرض .

٢ - أحكام التجويد :

في حالة اختيار الرقم ٢ من اللوحة الرئيسية (أحكام التجويد)، تعرض لوحة جديدة خاصة بأحكام التجويد (شكل ١٣) . ومن الاختيارات الثلاثة ، يستطيع المستخدم إدخال إحدى هذه الأرقام . والرقم ٢ يقوم بالخروج من هذه اللوحة إلى اللوحة الرئيسية .

ادخل رقم الحزب : ٦٤ -

(شكل - ٩)

ادخل رقم الجزء : ٣ -

(شكل - ١٠)

ادخل اسم السورة : عيسى

ادخل رقم الآية : ٥٠

ادخل اسم الصورة :عبس_

(شكل - ١٢)

لوحة أحكام التجويد

١ عرض الحكم للآية معينة

٢ تعريف بعض أحكام التجويد

٣ الخروج

الاختيار :-

١-٣ استخراج الحكم لآية معينة :

عند اختيار الرقم ١ من لوحة أحكام التجويد ، يقوم النظام بطلب اسم السورة ورقم الآية كما في الفقرة (٢-٣) . ثم تظهر شاشة جديدة تخير المستخدم إذا أراد عرض الأحكام مفصلة أو أحكام أساسية والرغبة في عرض الأحكام واحدة بعد الأخرى من غير أن يضغط المستخدم على أية مفتاح . ويقبل الجهاز الحرف (ن) لنعم فقط عند الرغبة في ذلك أو أية حرف آخر لعدم الرغبة (شكل ١٤) .

بعد ذلك يقوم النظام بعرض الآية على الشاشة وبتحليل النص في الآية لأحكام التجويد . ويعرض الحكم في أسفل الشاشة حتى نهاية الآية . ومن ثم يمكن الإستمرار في استخراج الأحكام للآية التالية بالضغط على مفتاح (PgDn) أو التوقف بالضغط على مفتاح (CR) (شكل ١٥) .

٢-٣ تعريف لأحكام التجويد :

في حالة اختيار الرقم ٢ من لوحة أحكام التجويد ، تعرض لوحة جديدة بعدة إختيارات (الشكل ١٦) . ويقوم النظام بعرض التعريف المحدد من المستخدم بإدخال إحدى الأرقام من اللوحة (٨-١) والرقم ٩ يقوم بالخروج إلى لوحة أحكام التجويد . وهذه هي الإختيارات :

- ١ - أحكام النون الساكنة والتنوين .
- ٢ - أحكام الميم الساكنة .
- ٣ - أحكام الميم والنون المشددين .
- ٤ - أحكام المد .
- ٥ - أحكام المتماثلين والمتقاربين والمتجانسين .
- ٦ - الصفات المتضادة .
- ٧ - الصفات التي لا ضد لها .
- ٨ - أحكام الوقف .

جميع الأشكال (ن):

عرض الأشكال تلقائياً (ن):

(v_H) الحبا ($\varepsilon =$) مكية جزء ω حزب ω

فَمَ كَلَّا تَتَلَوْنَ (٥)

إظهار الغنة و الشدة

١ أحكام النون الساكنة و التنوين

٢ أحكام الميم الساكنة

٣ أحكام الميم و النون المشددين

٤ المد

٥ الحركات المتضادة

٦ الحركات التي لا أخذ لها

٧ إدغام المتماثلين و المتجانسين و المتقارب

٨ الوقف

٩ الخروج

الاختيار :-

عند اختيار أحد هذه الأرقام ، تظهر شاشة جديدة توجد بها التعريفات الخاصة بهذا الحكم . وباستخدام المفاتيح CR و PgDn يمكن للمستخدم التوقف أو الإستمرار في عملية عرض التعريفات . وفي نهاية التعريفات يرجع النظام إلى اللوحة السابقة حتي يتم اختيار رقم ٩ للخروج إلى لوحة أحكام التجويد . (شكل ١٧) يبين إحدى هذه التعريفات الموجودة في النظام .

وفي النهاية يتضح للمستخدم أن النظام سهل الإستعمال وكل شيء موضح أمامه من خلال الشاشات . وهذه الشاشات تحتوي على لوحات باختيارات متعددة منها الخروج الذي يرجع للوراء خطوة واحدة .

ونرجو لك التوفيق في الإستخدام والإستفادة التامة من النظام .

أحكام النون الساكنة و التنوين

الظهار : لغة البيان

إصطلاحاً إخراج كل حرف من مخرجه من غير غنة
 في الحرف المظهر إذا جاء بعد النون الساكنة
 أو التنوين أحد حروف الظهار =
 وحروفه ستة هي أ ه ع غ ج خ =

الادغام : لغة الإدخال و المزج

إصطلاحاً التقاء حرف ساكن بحرف متحرك بحيث
 يصيران حرفاً واحداً مشدداً من جنس الثاني إذا
 جاء بعد النون الساكنة أو التنوين أحد حروفه

Appendix B
USER MANUAL FOR THE DATA
ENTRY PERSONNEL

بسم الله الرحمن الرحيم

نظام لعرض وتخزين واسترجاع القرآن الكريم

دليل المستخدم

لمدخلتي المعلومات

بسم الله الرحمن الرحيم

نظام لعرض وتخزين واسترجاع القرآن الكريم

المقدمة :

يعتبر هذا الدليل بمثابة المرشد للأشخاص الذين يرغبون باستخدام هذا النظام . ويمكن استخدام هذا الدليل من قبل أي شخص بغض النظر عن مدى معرفته بالحاسبات الآلية . واستخدام النظام سهل للغاية بحيث يمكن لبعض المستخدمين استعماله من غير قراءة هذا الدليل .

ويتميز النظام بما يلي :

- ١ - عرض نصوص القرآن الكريم بالرسم العثماني .
- ٢ - إمكانية البحث عن كلمة ، أو مقطع من آية ، أو حزب ، أو جزء ، أو آية في سورة أو سورة .
- ٣ - عرض الأحكام المتعلقة بالتجويد في آية آية في القرآن الكريم .
- ٤ - إمكانية طبع النص القرآني على الطباعة .
- ٥ - استخدام النظام باللغة العربية .
- ٦ - سهولة استخدام النظام .
- ٧ - استخدام النظام على أجهزة أي بي إم الشخصية أو المطابقة لها .

ويتألف الدليل من أجزاء مختلفة يمكن قراءة أي قسم على حدة ما عدا القسم الخاص بكيفية البدء بتشغيل النظام الذي يجب أن يقرأ في البداية .

بدء تشغيل البرنامج :

يمكن استخدام الأسطوانات القاسية (Hard disk) لتسريع عملية حفظ المعلومات وذلك بنقل البرنامج من الأسطوانة الممغنطة (Floppy disks) .
راجع الدليل الخاص بالـ (DOS) لمعرفة ذلك .

ادخل الأمر التالي لتشغيل برنامج إدخال المعلومات :

A>EDKHAL

ستظهر لك الشاشة الموجودة في الشكل ١ ، التي تعتبر اللوحة الرئيسية للبرنامج .

اللوحة الرئيسية :

تحتوي اللوحة الرئيسية على أربعة اختيارات كما هو موضح في (الشكل ١) . وبإدخال إحدى الأرقام (١-٣) يقوم البرنامج بالانتقال إلى شاشة جديدة . أما الرقم ٤ فهو عبارة عن الخروج من البرنامج . عند الرغبة بإدخال المعلومات وحفظها يجب اختيار الرقم ١ . الرقم ٢ ، يعطيك الإمكانية لحذف ملف معين (جزء كامل) . أما الرقم ٣ ، فيمكنك من التغيير في بعض المعلومات عند اكتشاف أي خطأ في عملية الإدخال .

١ - الإضافة إلى ملف :

عند اختيار الرقم ١ من اللوحة الرئيسية (الإضافة إلى ملف) ، ستظهر عبارة (ادخل رقم الجزء المطلوب) أسفل الشاشة (شكل ١) . وعند ادخال الرقم من (١-٣٠) ، سيقوم البرنامج بالبحث عن الملف الموجود به هذا الجزء . وعند عدم التمكن من وجوده ، ستظهر عبارة تخبرك بأن (الملف جديد) . ثم تظهر شاشة جديدة يمكن من خلالها إدخال المعلومات (شكل ٢) .

اللوحة الرئيسية

١ : الاضافة إلى ملف

٢ حذف ملف

٣ التغيير في ملف

٤ الخروج

الاختيار ١:

ادخل رقم الجزء المطلوب : ٥-

لوحة إدخال المعلومات

الله أكبر

(شكل - ٢)

HEX	ENG	ARB	HEX	ENG	ARB	HEX	ENG	ARB	HEX	ENG	ARB
01	^A	ٲ	37	7	٧	4F	O	د	67	g	هـ
02	^B	ٲ	38	8	٨	50	P	ذ	68	h	و
05	^E	ٲ	39	9	٩	51	Q	ر	69	i	ى
21	!	---	3A	:	:	52	R	ز	6A	j	ي
22	"	و	3B	;	---	53	S	س	6B	k	---
23	#	ل	3C	<	---	54	T	ث	6C	l	---
24	\$	---	3D	=	=	55	U	ص	6D	m	---
25	%	٥	3E	>	∴	56	V	ض	6E	n	---
26	&	٤	3F	?	---	57	W	ط	6F	o	---
27	'	---	40	@	ل	58	X	ظ	70	p	---
28	()		41	A	ـ	59	Y	ع	71	q	---
29) (42	B	ٲ	5A	Z	غ	72	r	---
2A	*	*½	43	C	أ	5B	[---	73	s	---
2B	+	*½	44	D	ؤ	5C	\		74	t	---
2C	,	*½	45	E	إ	5D]	---	75	u	---
2D	-	-	46	F	و	5E	^	---	76	v	---
2E	.	---	47	G	ا	5F	_	---	77	w	---
30	0	٠	48	H	ب	60	٠	---	78	x	---
31	1	١	49	I	ة	61	a	ف	79	y	---
32	2	٢	4A	J	ت	62	b	ق	7A	z	---
33	3	٣	4B	K	ث	63	c	ك	7B	{	---
34	4	٤	4C	L	ج	64	d	ل	7C		---
35	5	٥	4D	M	ح	65	e	م	7D	}	---
36	6	٦	4E	N	خ	66	f	ن	7E	~	*سج

أنها فراغ . وبقية العلامات تعتبر حركات . وعند حدوث أي خطأ في الطباعة ، يستخدم مفتاح الحذف (—) لحذف الحرف الأخير وحركته معاً ومن ثم يمكن إدخال المعلومات الصحيحة مرة أخرى .

عند الإنتهاء من إدخال آية كاملة يقوم المستخدم بالضغط على مفتاح العودة (CR) ومن ثم يقوم النظام بمسح المعلومات المكتوبة على الشاشة وتظهر عبارة تسأل المستخدم الإستمرار أو الإنتهاء من عملية الإدخال . إذا ضغط المستخدم على الحرف (ن) يعني (نعم) فإن البرنامج سوف يخرج من هذه الشاشة ويعود إلى اللوحة الرئيسية. وإلا فإنه سوف يعود إلى أعلى الشاشة ويقوم بالعملية نفسها للآية التالية (شكل ٣) .

٣ - حذف ملف :

عند اختيار رقم ٢ من اللوحة الرئيسية (حذف ملف) ، ستظهر عبارة (ادخل رقم الجزء المطلوب:) أسفل الشاشة (شكل ١) . يقوم المستخدم بإدخال رقم بين (١-٣٠) . وسيقوم البرنامج بحذف الملف إن وجد ويعطي عبارة (الملف حذف) ليبين ذلك . وإلا فإنه سيعطي عبارة (الملف غير موجود) إذا لم يجده ويرجع إلى اللوحة الرئيسية.

٣ - التغيير في ملف :

إذا اختار المستخدم الرقم ٣ من اللوحة الرئيسية (التغيير في ملف) فإن بإمكانيته أن يغير كلمة أو عدة كلمات بحيث لا يكون طولها أكبر من ٢٠ حرفاً . ويجب أن يدخل المستخدم الكلمة بحركاتها صحيحة كما هي مخزنة ثم يدخل الكلمة الجديدة بحركاتها أيضاً .

عند اختيار الرقم ٣ ستظهر شاشة جديدة . ويطلب البرنامج من المستخدم إدخال رقم الجزء . وإذا لم يوجد الجزء المطلوب فسوف

لوحة إدخال المعلومات

نهاية إدخال المعلومات نعم : _

(شكل - ٣)

تظهر عبارة (الملف غير موجود) في أسفل الشاشة ويعود البرنامج إلى اللوحة الرئيسية . ثم يطلب البرنامج رقم السطر الموجودة به الكلمة. وكذلك الكلمة المراد تغييرها والكلمة الجديدة (شكل ٤). عند إدخال الكلمة فإنه يتبع نفس العمليات المذكورة في الاختيار رقم ١ . ثم يقوم النظام بعملية التغيير المطلوبة ويعود إلى اللوحة الرئيسية .

وفي النهاية ، نرجو لك التوفيق في استخدام البرنامج ونرجو أن تجزى الجزاء الوفي في عملك هذا إن شاء الله تعالى . وجزاك الله خيراً .

- 1 أدخل رقم الجزء المطلوب :
- 1 أدخل رقم السطر :
- 1 أدخل الكلمة المراد تغييرها : ^{س٢}الَّذِينَ
- 1 أدخل الكلمة الجديدة : ^{س٢}الَّذِينَ